

ENHANCING IDENTIFICATION AND REPORTING OF POTENTIALLY HARMFUL PUBLIC DATA ON DANISH ORGANIZATIONS

by

Rasmus Lau Petersen

Supervisor: Christian Damsgaard Jensen



M.Sc.Eng.IT master thesis

DTU Compute
Technical University of Denmark
Kongens Lyngby, August 1st, 2017

Summary (English)

This master thesis aims to aid and enhance the current processes used to identify and report in *Open Source Intelligence* (OSINT) on Danish organizations.

This is data generated by the daily work of the organization and its employees when they act and communicate. The data is collected by public registries or commercial 3rd parties, which can provide a valuable source of information for an attacker intending to target organizations. Security professionals are aware that such data exist and report on it as part of their services, but collecting it effectively is difficult. Relating it to the domain of the organizations acting under the different Danish legislation and standards can be difficult. The organizations themselves may have a good overview of the latter, but lack overview and awareness of OSINT and the attack scenarios this enables.

We attempt to enhance the process of identification and reporting in two ways: By developing plug-ins (“transforms”) for the widely used OSINT-gathering program Maltego by Paterva and by developing a framework for inputting findings from Maltego to generate a report categorizing findings and relating them to common OSINT-enabled attack scenarios and applicable legislation, standards and guidelines.

We examine current methodologies for conducting vulnerability- and penetration test assessments, standards, guidelines and Danish legislation pertaining to OSINT-data and identify and analyze a dozen common OSINT-enabled attack scenarios. The section on standards finds valuable, general guidance to enhance security maturity in organizations, but only little pertains to hinder OSINT-generation.

Traits of social engineering-techniques, which often are involved in these types of attacks, are also analyzed.

Two transforms were successfully developed for the Danish domain registry DK Hostmaster and a commercial supplier of aggregated data from the registry of Danish vehicles and debt of these. They can be included in and enhance the work processes in a security consultancy. The development also highlights problems with developing for closed source-software; due to the difficulties faced and solved in regards to this, a section on developing transforms for Maltego is included in the appendix.

Based on the examined legislation, standards and guidelines and the scenarios created, a report-

generation framework has been developed. It takes an export from Maltego and by manually assigning labels to each entry, outputs a report. The report imitates examples of commercially used reports using colors and summaries to make it easy-read. It connects the findings to the scenarios describing specifically targeted attacks and three of the surveyed standards.

The framework works and outputs successfully as-is, but the work highlighted difficulties with linking the domains of data labels of actual findings to standardized scenarios and formal standards. It is an essential task to get these links correct to ensure proper conclusions in the output, so the report can be used as-is in the product portfolio of e.g. a security consultancy.

We list suggestions for future work in the discussion and conclusion and highlights the need for conducting the research with input from e.g. security professionals of consultancies and internal organizational security functions.

Summary (Danish)

Målet for denne afhandling er at støtte og forbedre processerne omkring identifikation og afrapportering af *Open Source Intelligence* (OSINT) der måtte eksistere om danske organisationer. Denne type data bliver genereret som et bi-produkt af processerne omkring det daglige arbejde organisationen og dens ansatte udfører, når de agerer og kommunikerer. Dataen opsamles af offentlige instanser og kommercielle tredjeparter og kan udgøre en værdifuld kilde til information for angribere, der måtte ønske at angribe organisationerne. IT-sikkerhedsprofessionelle er opmærksomme på eksistensen af denne type data og rapportering af den indgår som en del af deres tilbudte services, men opsamlingen af den er besværlig og tidskrævende, herunder også relatere dataen til den virkelighed organisationerne agerer under inkl. dansk lovgivning og standarder. Mens organisationerne nok har et godt overblik over sidstnævnte, kan de mangle indsigt i hvad OSINT-data er og de angrebstyper der er forbundet herved.

I projektet søger vi at forbedre processerne for identifikation og afrapportering på to måder: Dels ved at udvikle plug-ins (“transforms”) til det vidt udbredte OSINT-indsamlingsværktøj Maltego udviklet af Paterva og dels ved at udvikle software, som ud fra resultaterne fra Maltego kan generere en rapport, som kategoriserer de gjorte fund og relaterer dem til almindeligt forekommende cyberangreb, hvor OSINT-data spiller en rolle samt gældende lovgivning, standarder og vejledning.

Vi undersøger aktuelle metoder til at udføre sårbarheds- og penetrationstests, standarder, vejledninger og dansk lovgivning som beskæftiger sig med OSINT-data. Endvidere identificerer og analyserer vi omkring et dusin typiske cyber angreb, muliggjort af OSINT-data. Afsnittet identificerer værdifuld, generel vejledning, som kan hjælpe med at forbedre sikkerheden i organisationer, men kun en mindre del af den beskæftiger sig med at forhindre at der genereres OSINT-data. Vi undersøger og beskriver også typiske træk ved “social engineering”-teknikker, som ofte bruges ifm. denne type angreb.

Der blev succesfuldt udviklet to transforms, som henter data fra dels den danske domæneregistrant DK Hostmaster og dels en kommerciel udbyder af et aggregeret datasæt over motorregistret samt gæld fra Tinglysningen ifm. dette. De to transforms kan inkluderes i arbejdsgangen hos IT-sikkerhedskonsulenter og forbedre arbejdsprocessen herved. Udviklingen fremhævede dog også problemerne ved at udvikle til closed source-software; pga. udfordringerne herved, er de undervejs i projektet identificerede løsninger repræsenteret særskilt i appendix.

På baggrund af den undersøgte lovgivning, standarder og vejledninger og de opstillede scenarier, blev der udviklet et stykke software til at generere en rapport. Softwaren kører på en eksporteret fil fra Maltego og ved manuelt at tildele typer til hvert stykke data i filen, genererer softwaren en rapport. Rapporten efterligner eksempler på kommercielt fremstillede rapporter ved at bruge farver og små konklusioner for at gøre den let forståelig. Den forbinder de i Maltego fundne data med de opstillede scenarier (dog kun målrettede angreb) og tre af de undersøgte standarder, som bedst lod sig relatere til OSINT-data.

Softwaren fungerer og genererer i den nuværende udgave, men det udførte arbejde tydeliggjorde problemer med at koble praktisk orienterede datakategorier med standardiserede angrebsscenarier og formelle standarder. Det er essentielt at kunne udføre disse koblinger korrekt, for at sikre, at konklusionerne i den automatisk genererede rapport stemmer bedst muligt overens med virkeligheden og kan bruges i produktporteføljen i et IT-sikkerhedskonsulentfirma.

Vi beskriver forslag til fremtidigt arbejde i diskussionen og konklusionen og fremhæver nødvendigheden af at udføre den videre research med input fra f.eks. IT-sikkerhedsprofessionelle fra både konsulenthuse og interne IT-sikkerhedsafdelinger ude i organisationerne.

Acknowledgements

I would like to thank all the people around for enabling this project.

In particular I would like to thank Keld Norman and Andy Cini for growing the initial idea with me and for Christian D. Jensen to join in on the idea and supervise the project for, including bi-weekly meetings, ad-hoc late night e-mails and valuable input.

I would like to thank *Specialklinikken Rebild* by J. Erling Pedersen-Bach and massage therapist Cathrine Thovtrup, without whose treatments it would have been next to impossible to write a master thesis with the whiplash/concussion I acquired three years back, but now finally are in recovery from.

I am grateful for my wonderful girlfriend whom have supported and helped me all the way through and especially when things got a bit busy. I look forward to spend much more time with you in our new apartment, which we also managed to buy during the month up till hand-in!

An extended thank you to Rosita Kanto, Frederik Eriksen & William Embarek for proof-reading my report and shout-out to all the good people on ST Nybrogård Kollegiet, whom have had to listen to my grumblings during up's and down's.

Last, but most certainly not least, where "thank you's" do not even suffice: A sincere, heartfelt "thank you" to my parents, Nina & John Petersen, for 27 years of love, support, blood, sweat and tears, which have enabled me to get to this point! I am most grateful and will never be able to pay it back enough.

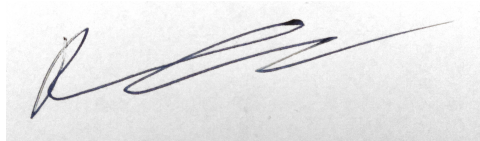
Preface

This thesis was prepared at DTU Compute in fulfilment of the requirements for acquiring an M.Sc. in Engineering and IT.

The thesis deals with the problems of organizations acting in an increasingly connected world where all actions leaves traces. To discover and understand the implications of data on open sources is difficult, but the goal was to enhance the process by developing new plug-ins to a popular platform for OSINT-investigations and a framework for automatically generating a report concluding on the findings and relating them to applicable legislation, standards and guidelines.

An overview of the thesis' content is found in Section 1.2.

Lyngby, August 1st, 2017

A handwritten signature in black ink, appearing to read 'Rasmus Lau Petersen', written on a light-colored background.

Rasmus Lau Petersen

Contents

1	Introduction	11
1.1	Problem definition	13
1.1.1	Goals	15
1.1.2	Risks	15
1.1.3	Work methods	16
1.2	Overview of this report	17
2	State-of-the-art	19
2.1	Penetration testing	19
2.1.1	Pre-engagement interactions	22
2.1.2	Intelligence gathering	24
2.1.3	Threat modeling	26
2.1.4	Vulnerability analysis	29
2.1.5	Exploitation	31
2.1.6	Reporting	32
2.2	Risk analysis	33
2.2.1	A general view: DS/ISO 27000-series	33
2.2.2	Specific methodologies	35
2.3	Standards and guidelines	38
2.3.1	DS/ISO/IEC 27000-series	40
2.3.2	CFCS (DK)	46
2.3.3	NIST (US)	50
2.3.4	CPNI (UK)	53
2.3.5	NCSC (UK)	55
2.3.6	Federal CIO Council (US)	57
2.3.7	Agency for Digitisation (DK)	60
2.3.8	Other sources	61
2.4	Social engineering-techniques	63
2.4.1	The attack-phases	63
2.4.2	The psychological tricks of a social engineer	66
2.5	Common OSINT-enabled attack scenarios	70
2.5.1	Targeted attacks	72

2.5.2	Un-targeted attacks	76
3	Analysis	79
3.1	Cyber security in a work environment	79
3.2	The role of external security consultants	80
3.3	Maltego transforms for Danish OSINT-sources	81
3.4	An auto-generated OSINT report	84
3.4.1	Categorizing the data for the auto-generated report	88
4	Design & implementation	92
4.1	Maltego transforms	92
4.1.1	Designing entities	94
4.1.2	DK Hostmaster-transforms	95
4.1.3	License plate-transforms	98
4.1.4	Implementation	99
4.2	The auto-generated report	101
4.2.1	Design of the output report	101
4.2.2	Designing the program	103
4.2.3	Implementation	105
5	Tests	113
5.1	Transforms	113
5.1.1	DK Hostmaster-transforms	114
5.1.2	License plate transform	122
5.2	Auto-generated report	134
6	Discussion	139
7	Conclusion	143
7.1	Future work	144
	References	146
A	How to develop Maltego transforms	152
A.1	Maltego basics	152
A.2	Making non-OEM transforms	153
A.2.1	Making custom entities	154
A.2.2	Distributing the transforms	155
B	Vulnerability reports	157
B.1	Qualys	157
B.2	Dubex A/S executive summary	164
B.3	Example of an auto-generated report	169

B.4 Example of a minimal, auto-generated report 186

C CPNI hostile reconnaissance checklist 202

D Figures 205

D.1 nrpla.de field implementation 208

D.2 Sample csv-export from Maltego 211

Chapter 1

Introduction

An “IT security incident” can take many different shapes. It can involve a direct attack by an adversary on a system exploiting his extensive knowledge of cryptography or specific soft-/hardware. It can also be an employee whom, with or without intent, gives access or discloses company data. Numerous solutions and techniques are employed to counter such incidents; examples are software like anti-virus engines, hardware like firewalls or IDS/IPS or managerial processes to audit and control. This has reduced the “exploitability” of many systems and organizations, where previously gain access through simple techniques. Due to these precautions, adversaries have to seek new ways to reach their goals (be it fortune, fame, disagreement, activism, espionage, commercial, diplomatic or economic advantage and anything in between [37]).

Social engineering has been employed through-out time, formerly by pick-pocketers, quack doctors and in general manipulating people¹, but in recent years it has successfully been used as highly viable technique in cyber security; in fact, social engineering is one of the biggest threats emerged in the recent years [2, 46], and it targets both users and organizations (i.e. everyone in the developed world).

Social engineering is a term used broadly to describe psychological manipulation of people to help or disclose information to an adversary within the context of computer security (see e.g. [37, 25]). In this way, the adversary bypasses all technological countermeasures that may be in place and directly exploits authorized and authenticated users (be it of some virtual environment or a physical location) to gain access or extract information. A human does not act deterministic in the same way as a piece of software and rules and procedures are often only final within well-defined processes. A human acts on several different other properties to make decisions and will factor in relations/power structures, appearance and subjective understandings of anything. A human can thus prove to be a much easier target for an adversary, because he might outright hand out the information queried for to the adversary if he believes the adversary are eligible! Eligibility can be established by exploiting some of the above factors and establishing a false context of the adversary or some situation.

¹An example of popular culture is the book and movie “Catch me if you can” of Frank Abagnale Jr. using social engineering to acquire sufficient knowledge to manufacture his own checks, fly for free and impose as a lawyer.

The social engineer acts opportunistic in an automated/manual and non-targeted/targeted process to discover and breach the security of individuals and organizations. He is driven by various motives will take any and every opportunity and bit of information to expand his knowledge of the target.

A social engineer might employ several different stages to increasingly built knowledge and relationships[35] leading to very elaborate schemes and surprising combinations of data. The capabilities and motivation of the attacker is almost infinite [37] (current examples are described in Sec. 2.5).

A wide range of adversaries exist with varying capabilities and goals. It ranges from the simple, manual attacks over automated attacks of varying complexity and ingenuity to state-sponsored groups with virtually unlimited funds, time, experience and a goal to conquer the world; groups of this caliber are typically also referred to as *advanced persistent threats* (APT's); they may target an individual or organization for several years to find the perfect opportunity [9].

Fortunately, a large number of the every-day attacks seen are not directly aimed at a single target; only high-value targets may hold assets of sufficient value to justify an APT attacking them. The rise in the threat of social engineering reported in [2, 46] is largely automated drive-by-like attacks.

Such attacks use publicly available information typically referred to as “*Open Source Intelligence*” (OSINT)² to create and target e.g. phishing attacks on individuals and organizations. OSINT is *all* publicly available information – it may be *footprints* of the organization and its employee’s daily operations (e.g. from public registers (government or 3rd party)), a product of use of IT systems, web content (e.g. articles, documents and their meta-data), news or active information sharing by individual employees on e.g. social media and fora. Some of the footprints are avoidable, some are not.

To find the information, the attacker can use search engines like Google and Shodan, but also the organizations’ own sites, government sites or public registries. The information found is then utilized to try to exploit human psychological mechanisms (i.e. social engineering) to e.g. establish context with the victims such that they place an unmerited degree of trust on an object/subject.

Social engineering can also be used as a way to collect information on the target to enhance a later attack through other means (e.g. malware, physical penetration).

Most of this information are pieces of data which the organization do not consider confidential, but do keep to themselves, e.g. mail addresses to stores in a chain and their managers [35].

²A definition can be found in [53], which is referenced by both NIST and NCSC, but it may be a bit strict and not adhere to all people’s perception of what OSINT is; Danish CFCS simply puts it as all public accessible data: <https://fe-ddis.dk/Opgaver/Efterretningstjeneste/Pages/Efterretningstjenesten.aspx>.

Managers are not too used to external correspondence on this e-mail, so when the attacker discovers these, he can deliver an attack with high credibility.

An example of an attack type is to, after having identified e.g. a C-level employee and maybe a partner/supplier, ask another employee for a transfer of money; this type of attack is often referred to as *spear-phishing* (i.e. a targeted phishing attack) [8] or *CEO-fraud*.

Another example of using OSINT, could be using an employee name and associated e-mail address to send e-mails to other employees to deliver a malicious payload or extract information like getting credentials for a website; this attack type is often referred to as *phishing*, but can lead to a range of other security incidents.

It is difficult to establish just exactly *what* information is at risk and should be withheld; the same piece of data on one platform might not have the same amount of risk on another platform. Technological measures are often also inadequate: *“Few effective technical security controls exist that can defend against clever social engineering attacks. Often the best solution is to provide periodic awareness and training of policy, guidelines, and best practices.”* [25]. Thus, to hinder such attacks, organizations adopt policies and procedures and promote awareness of the issues. The organizations *“[...] have no control over [the attackers’] capabilities and motivations, but [...] can make it harder [...]”* or impossible for the attackers to gather the required information, because *“[w]hilst attackers may have the capability and the motivation, they still need an opportunity to deliver a successful attack.”* [37].

Recommendations often includes minimizing unnecessary exposure of corporate structure, key individuals (e.g. in HR, accounting or IT services), partners, procedures, employees in general (including private information about them) etc.

Data leaks can happen virtually *anywhere*. In this increasingly connected world, where a large part of the life and activities of individuals and organizations (and their employees) is happening online in some way, the task of maintaining an overview of all this is proving very difficult. To help identify these leaks of information, organizations usually employs security professionals/researchers to perform *penetration tests* (“pen-test”) or vulnerability tests against the customer in a manner similar to the adversary. This is a manual, time-consuming task highly depending on the researcher’s knowledge and imagination.

1.1 Problem definition

The aim of this thesis is to provide tools to support the common task of security researchers evaluating/testing the security of organizations; here for Danish organizations. The “security” to be tested is specifically the exposure of the client organization by OSINT-data.

We will survey the context in which such tools will enter into in “the real world” and what requirements need to be setup to enable this.

From this, we will seek toThe scenarios are based around modern cyber-crime methods ((spear-) phishing, ransomware, APT’s (with e.g. political or economic motives) and other OSINT- and social engineering-techniques).

The tools are to be built as *transforms* for Maltego³. It is a well-known, de-facto industry standard program for aiding security assessments (e.g. pen-tests), especially for case management of pen-tests and for performing passive reconnaissance on target organizations or individuals by both security researchers and adversaries. The transforms⁴ should be distributed through the official “hub” integrated into the Maltego GUI; they are thus recognized and usable by most researchers⁵.

The auto-generated report should automate part of the report creation process, which is an integral part of a commercial pen-test, to provide results in a uniform way and demonstrating violations of regulations or standards as applicable. It should be usable to the researcher to guide the organization on how to reduce their attack profile by showing critical data found in the pen-test.

This typically includes evaluating the organization’s compliance with current standards and/or guidelines (scoped with respect to primarily Danish organizations, but will also include reputable international sources) as well as use cases of current attackers and their methods searching for essential information to gain privileged access to an IT-system or the trust of their target to exploit them. The overview should be structured in a way such that the researcher/organization can readily identify where to best spend their man hours (in accordance with their own risk assessment and relevant standards and regulations) in order to mitigate OSINT- and social engineering-enabled attack types.

It should be easy to interpret for IT-professionals.

Data input for the auto-generated report could come from many sources, which requires a lot of development time. To make use of the fact that there are transforms already being developed for the Maltego platform, the report input can come from there too. Both data gathering and reporting can thus be done using a known platform such that any capable IT security professional can perform the analysis. This will allow IT security consultant companies and other professionals to use the products developed in this thesis.

The collection of OSINT is already a discipline integrated in many platforms, so emphasis here is put on the large number of publicly available sources of data in Denmark being made available for free as a part of the strive to induce economic growth⁶ and Denmark’s commitment to the G8-countries *Open Data Charter*⁷. This is to my best knowledge a novel approach to include queries towards these sources⁸.

³The choice is further discussed in Chapter 3.

⁴May consist of a custom configuration, entities (data types) and transforms (functions to perform some action on an entity), but often called *transforms* as a whole in the Maltego documentation

⁵Guidance and tutorials are widely available online in both video and text if the security professional is unfamiliar with Maltego.

⁶See e.g. <http://www.opendata.dk/om/hvad-er-open-data-dk>

⁷Acceded June 18th 2013: <https://www.digst.dk/ServiceMenu/Nyheder/Nyhedsarkiv/Digitaliseringsstyrelsen/2013/Open-Data-Charter>

⁸An e-mail conversation with the Head of Development at DK Hostmaster medio March 2017 supported this view.

It is acknowledged that IT security professionals have varied approaches to performing vulnerability assessments/pen-tests like the above. This is sought countered by consulting a variety of different sources for the methods and standards, selecting the most evident data sources, best practices and informal, continuing conversations with IT security professionals I have the privilege of working together with.

1.1.1 Goals

In short, this master thesis delivers the following products (besides this report):

Product	Description and goals
Maltego seed with transforms and custom entities	A complete, production-ready seed to add to the transform hub of Maltego of transforms for querying a range of Danish OSINT-sources relevant for acquisition of information on Danish organizations and necessary entities in accordance with Maltego developer guidelines such that the transforms can be used in combination with other activities of the pen-test performed in/with Maltego. The transforms should in particular help to reduce part of the manual work to gather OSINT from Danish sources on the client organization.
Auto-generated report	A report auto-generated from the findings of an investigation (e.g. a pen-test) conducted in Maltego. The report compares the findings to common cyber attack scenarios, relevant legislation and standards (Danish context) for the researcher to use in the final pen-test report to be delivered to the customer. In particular the report enhances this task by linking the findings to the relevant scenarios and legislation/standards and outputting the results in a well-formed report.

Table 1.1: Product specification of the products of this thesis.

The requirements of each of the two deliveries are based on an analysis carried out in Chapter 3 and put in detail there.

1.1.2 Risks

While it is not possible to know all aspects of the risks initiating a larger project like this, it is important to consider what kind of risk there are, how they may affect the project and mitigative steps. Before project initiation we identified the following risks:

- The products are open-ended which can make the work go off track. A direction of the work should be established early by examining relevant sources to guide further process.

- How can we automate a manual report-generation? Developing can possibly be done, but understanding the data behind may require some intelligence.
- The availability of adequate sources and offered API's for Danish OSINT-sources are important to the project. A lack hereof has to be discovered early to allow for different possibilities to be explored (e.g. international sources).
- Developing for Maltego is a new field to the author. We do not know which possibilities exist for exporting besides regular pdf-files and how we can interface with the program at all.
- What is the availability of standards/guidelines pertaining the specific types of data outside the regular assets-models that we have met in literature on the university? How does guidelines look when they are not directing on how to update the organization's systems?

The risks are concluded on in the conclusion of this report (Chapter 7).

1.1.3 Work methods

To form an effective process for the author, the work was planned as 2-week "sprints" (like used in the Scrum software development-method) with bi-weekly supervisor meetings to update on the current work and discuss what work to be carried out in the following 2-week period. Online tools to manage the work process were experimented with using `trrello`-boards⁹ to maintain a backlog of tasks and assignment to each sprint, but for a single-person project, this was an unnecessary work-load to keep updated through-out the project. It was however an intriguing process found very usable for cooperating with others.

To discover papers for the state-of-the-art chapter, we went through literature presented in previous courses as well as resources from government cyber security entities and news articles. We also considered standards and guidelines from Danish and international official sources, which are all sources we have either had referenced as part of the lectures, in related material or in the study job.

The report was typeset with \LaTeX and edited in `Texmaker`¹⁰, which again proved to be a two-edged sword offering both easy, straight-forward writing and numerous difficulties with getting a few exotic features just right. It does look nice though.

Versioning and backup were done using the DTU-provided `subversion` servers and the `TortoiseSVN`-client¹¹. It proved useful a couple of times to revert slippage in coding and editors.

The work flow has been steady and consistent, but in the early phases it was influenced by my whiplash/concussion having now passed its three years "jubilee". After taking 1¹/₂ week out of the project to get a treatment, this improved somewhat.

⁹<https://trello.com>

¹⁰<http://www.xmlmath.net/texmaker/> – just released a new version.

¹¹<https://tortoisesvn.net/>

I am pleased with the bi-weekly meetings with my supervisor, Christian D. Jensen, which helped to keep focus on the assignment and resolve blocking issues.

1.2 Overview of this report

This thesis is structured as follows:

- To give a baseline for the work and to be able to show in what way this thesis improves the current situation, Chapter 2 demonstrates contemporary theory and methods of the following subjects:
 - Penetration testing (Sec. 2.1), to demonstrate how an adversary (e.g. a social engineer) or a security researcher in a structured way will identify targets and find the necessary information to leverage his attack.
 - Risk analysis (Sec. 2.2), to show how we currently try to understand, formalize and mitigate how adversaries will attack organizations.
 - Standards and regulations of data handling/leaks (Sec. 2.3), in particular concerned with the control of data flowing to/from public sources. By relating to these later, we show an urgency of the findings by violation of a standard or law. Emphasis is put on Danish organizations and applicable regulations and guidelines, but also includes larger government bodies as the National Institute of Standards and Technology, US (NIST) and The National Cyber Security Centre, UK (NCSC).
 - General description and background of common OSINT-enabled cyber attack types (Sec. 2.4), including an overview of the psychological tricks a social engineer can employ (Sec. 2.4.2).
 - Common OSINT-enabled cyber attack scenarios presented by examples (Sec. 2.5) to get concrete evidence of adversaries' attack vectors. Emphasis on what OSINT-data were required to effectuate the attacks.
- Chapter 3 analyses the contemporary methods and theory of the previous chapter to understand where an organization and its cyber security contractors could improve and how this can be done using the products of this thesis (see deliveries in Section 1.1.1). The Chapter also identifies a selection of Danish sources to gather OSINT from.
- Chapter 4 is concerned with the actual design of the Maltego transforms/entities and the auto-generated report to fulfill the requirements of Chapter 3. It also describes the concrete implementation of the Maltego transforms and the auto-generated report and the considerations.
- Chapter 5 demonstrates how the developed deliveries meet the requirements put forward in Chapter 3 by relevant tests.

- Chapter 6 discusses problems and discrepancies in the project and in particular between the requirements and the final product. Insights are offered on the issues identified during the process, highlights the projects contributions and engages the reader to understand the issues found.
- Chapter 7 concludes on the findings and results of the project in relation to the problem definition and -domain.
- The appendix contains a section on developing for Maltego with advice and links to the proper parts of the documentation (App. A), examples of vulnerability reports (App. B) and samples of our own (App. B.3 and B.4), the CPNI checklist [11] (App. C) and a section of miscellaneous figures (App. D).

Chapter 2

State-of-the-art

This chapter aims to provide a baseline for the rest of the thesis by collecting information on the various subject related to it. Motivation for each section is found in the beginning of it and in the general overview of the report in Sec. 1.2.

The chapter demonstrates contemporary methods for:

- Penetration testing including vulnerability scans (Sec. 2.1).
- Threat modeling and risk analysis (Sec. 2.2).
- Current legislation, standards and guidelines for data handling with special regards to data that may be public (Sec. 2.3). Emphasis are put on Danish organizations and applicable regulations and guidelines and reputable sources.
- A general description of common OSINT-enabled attacks and tricks exploited by social engineers (Sec. 2.4).
- Current examples of common types of OSINT-enabled attacks (Sec. 2.5), some leveraging the techniques described in the previous section.

2.1 Penetration testing

This section gives an overview of the current state-of-the-art methods and tools in the domain of penetration testing in the phases of information gathering, threat modeling, vulnerability analysis (with emphasis on social engineering) and other things relevant to this thesis. Other phases of a pen-test are shortly touched upon.

The aim of the section is to get an introduction and understanding of the context in which the deliveries of this thesis are expected to be part of.

It should be noted, that while effort has been put in describing state-of-the-art, law enforcement across the world possesses some quite capable tools within this domain (as evident from e.g. the

Snowden leaks¹, but they are also adamant to not disclose information about them! The below should thus be considered “state-of-the-art methods for commercial security researchers”.

The aim of a pen-test is to “*Identify ways to exploit vulnerabilities to circumvent or defeat the security features of system components.*” [45]. The time frame of a pen-test is from days to weeks depending on the scope.

This is opposed to the vulnerability scan, which “*Identify, rank, and report vulnerabilities that, if exploited, may result in an intentional or unintentional compromise of a system.*” [45] and typically takes only minutes per host.

A pen-test typically includes a vulnerability scan to identify vulnerabilities, but goes even further to test if one or more of these are exploitable. Thus a pen-test is also an opportunity for the client organization to test their procedures including emergency response plans.

In an industry, where every actor can claim a title as “pen-tester” and work methods are very diverse and personal, standards are not widespread.

Effort has been put to identify specific sources, which seems to present content comparable to other source and referenced widely. Some exist as a part of course material within the large providers², but are protected and sold as course material only.

As a response to the lack of open standards, an open, community developed standard has been made: The Penetration Testing Standard (PTES) [56]. This standard is referenced in a widely popular book [61], is recommended by InfoSec Institute³ and is the basis of the GIAC-material⁴ and [30]. In my own opinion, [56] seems very comprehensive and covers everything what has been taught at DTU.

Finally, I have consulted the *Payment Card Industry Data Security Standard’s* (PCI DSS) penetration testing guidelines [45]. It serves as supplementary information to organizations adhering to the PCI DSS requirements (typically obligatory for organizations handling credit card payments). This guide suggest other sources for a pen-testing framework; this includes PTES and notably also *NIST Special Publication 800-115* [49] and the *Open Source Security Testing Methodology Manual* (“OSSTMM”)⁵ which one can also survey if so wished. It would be redundant to mention these throughout this section too, as they do not noticeably differ from the already chosen frameworks.

The stages and the actions in each step of a pen-test, can hard to put in exact terms as per the above. According to [56], the phases of a full, professional pen-test are:

¹See e.g. https://en.wikipedia.org/wiki/Edward_Snowden#Global_surveillance_disclosures or big data analysis tools like *Palantir Gotham* (used in e.g. the Danish police force): https://en.wikipedia.org/wiki/Palantir_Technologies#Products

²E.g. GIAC (<https://www.giac.org/>) and (ISC)² (<https://www.isc2.org/>). I have identified a paper detailing penetration testing methodology [59] from the SANS Reading Room, which is a resource for content of good quality written by people related with the SANS Institute training.

³See <http://resources.infosecinstitute.com/penetration-testing-intelligence-gathering/>

⁴I was told so by two colleagues having passed “GIAC Security Essentials” (GSEC) – equivalent to (ISC)²’s “Certified Information Systems Security Professional” (CISSP).

⁵<http://www.isecom.org/mirror/OSSTMM.3.pdf>

1. Pre-engagement interactions
2. Intelligence gathering
3. Threat modeling
4. Vulnerability analysis
5. Exploitation and post-exploitation
6. Reporting

The phases are also depicted in Figure 2.1.

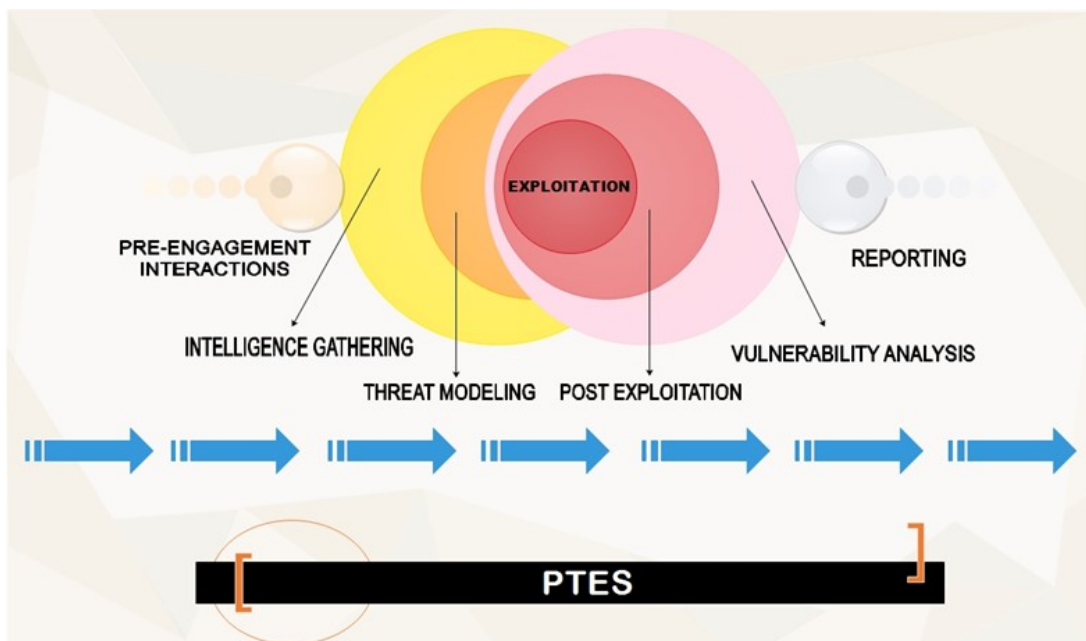


Figure 2.1: The phases of PTES. From <http://resources.infosecinstitute.com/penetration-testing-intelligence-gathering/>

The phases are virtually identical to those in [59], although *threat modelling* is not emphasized as a part of their methodology. It can however add great value for the organization to be able to rate the identified vulnerabilities in the end against the likelihood of them happening; to do this, it is necessary to know what kind of attackers may exploit a given vulnerability.

[45] uses a less fine-grained listing: Pre-engagement, Engagement, Post-engagement and Reporting. The contents of each phase is similar to the two others guides, there are just more actions to perform within each phase instead. We provide a better understanding and overview of the individual components of the test here by assessing them in smaller steps; the contents of [45] are however still included in each of the above phases as relevant.

[49] is not different from the three others and can be used by the reader if different formulations of specific subjects are needed; it is referenced some places in the following, where it provides better guidance.

The phases of intelligence gathering and vulnerability analysis are especially interesting to this thesis, as it is necessary to see how the phases can be performed currently and how the aimed product of this thesis (the extended tools for gathering OSINT on from Danish sources and an automated report on the value of the acquired data to an adversary) can complement the current available tools.

2.1.1 Pre-engagement interactions

This phase consists of planning actions and negotiations of details in the engagement between the pen-tester and the organization (the client). During this phase, the parties meet and agree on targets (if only specific parts of the network should be tested), degree of engagement, degree of interference and other rules of engagement. The client organization's perceived purpose of the test may also be aligned with the pen-tester (e.g. benchmarking, mitigation of identified vulnerabilities, meeting requirements of certain standards and what point-of-view mitigations will be judged from (cost/benefit vs. risk)) [49].

Knowledge of this is not relevant to this thesis, so it is only briefly touched upon.

2.1.1.1 Amount of information disclosed

The amount of information disclosed from the client to the pen-tester in this phase is a trade-off between how "realistic" the scenario should be and the time and cost consumption. The agreement with the client may also hinder the pen-tester from going to the same extent as a "real" attacker during the pen-test [59]⁶.

The types of information shared can be on specific employees and their data (including passwords to act as them [45]), departments or hardware/software/physical sites to target. We distinct the amount of information shared between the client and the pen-tester with the following notions (after [45]):

White-box testing: Testing performed with knowledge of the internal structure/design/implementation of the object being tested.

Grey-box testing: Testing performed with partial knowledge of the internal structure/design/implementation of the object being tested.

Black-box testing: Testing performed without prior knowledge of the internal structure/design/implementation of the object being tested.

2.1.1.2 Degree of engagement

The parties also agree on a "degree of engagement". This only exist explicitly in [56] and the notion of levels is something I have not seen earlier, but it makes good sense to graduate the scope (and time consumption) of the pen-test and do so in a standardized way. The levels are

⁶Even for great pen-testers with lots of time for the engagement, their capabilities will often lack behind state-sponsored groups.

subsequently used to determine which actions should be carried out in the later phases following PTES.

PTES shows this as part of the *intelligence gathering* phase, but following the flow of other standards, it is more relevant to have it as part of the re-engagement interaction phase, as the client may want a say on this.

The three levels of degree of engagement “depth” are⁷:

Level 1 *Compliance driven*; a “[...]click-button information gathering process”, which can be done almost entirely by automated tools.

Level 2 *Best practice*; level 1 and some manual analysis, where e.g. knowledge of the organizational domain, its partners, location, hierarchy is necessary.

Level 3 *State sponsored*; “red team, full-scope”; level 1 and 2 and a lot of manual analysis. Includes creation of social media profiles, on-location gathering (e.g. “dumpster diving”), analysis and establishment of real-world relationships on e.g. social media.

The levels are referred to in later parts of PTES where they are used to select tools.

2.1.1.3 Degree of interference with the target

Finally, the degree of interference should be established with the client. PTES distinguishes between three degrees of interference with the target during the information gathering phase:

Passive When the key point is to avoid attention to the information gathering process; the information should be collected without interfering directly with the target, so it shall use only archived information (which may be out of date or incorrect). Some tools exist for this (e.g. search engines, PGP key servers, web archives/databases of host names) so it is possible to perform.

Semi-passive Information gathering shall not be distinguishable from regular Internet traffic and shall not draw attention to the pen-tester. We only query published information and directories (i.e. not trying `domain.com/login.php` unless listed on the web page). Some scanning tools (e.g. `p0f`) claim to scan indubiously.

Active Will probably be detected by the target. Active mapping of the full network range and full port scans, enumerating services, directories, files, servers etc. Most activities will fall into this category in a normal pen-test.

Passive reconnaissance is a common term; PTES has it formalized a bit further than usual, which I think is useful for understanding the term.

⁷http://www.pen-test-standard.org/index.php/Intelligence_Gathering#Background_Concepts

2.1.1.4 Rules of engagement

It is important for the pen-tester to cover himself legally. [45] lists many points to consider, e.g. “*During what time window will testing need to be performed?*”, “*Are there security controls that would detect or prevent testing?*”, “*If equipment owned by the tester is to be connected to the organization’s network, what steps must be taken to ensure the equipment does not pose a threat to the environment[...].*”, and similar. This is not a great focus of the two other guides, by the three combined can give an insight to the interested.

2.1.2 Intelligence gathering

Intelligence gathering is the process of acquiring information on a target using open sources (often referred to as “Open Source Intelligence” or OSINT), on-/off-site gathering, “human intelligence” (HUMINT – intelligence acquired through human interaction; can typically not be acquired in any different way) and foot-printing of the organization (networks, systems, software, servers etc.). The aim is to gather as much information as possible in order to have a large amount of attack vectors during the later phases of the pen-test.

Depending on the specific pre-engagement deals made with the client organization, some information may be released prior to the test itself (see Sec. 2.1.1).

2.1.2.1 Identification of targets

Identifying the targets to gather data from is typically an integrated part of this phase, where discovery of data further enhances the search of new targets or may disclose targets directly to the researcher. However, some scoping may be known or have been decided beforehand (e.g. specific TLD’s, departments or employees).

2.1.2.2 Tools

The tools for performing the information gathering are numerous and comes in addition to the physical gathering, which can include visiting the organizations physical locations and e.g. observing and reviewing the security measures. It is not possible to list these exhaustively, but below an overview of the actions to take and the tools one could use is given.

There’s a *very* wide range of HUMINT and OSINT sources to gather intelligence from; from most, automated tools cannot directly aid the researcher unless he is searching specifically. An example is that the researcher can probably readily find important company dates (e.g. jubilees) on the company web page or LinkedIn, but no automated tool exists to collect these; how should they be distinguished from other dates (e.g. date of publishing of news posts)? Another example could be information on the new product line.

The context is important and while AI’s have come a long way, they have not been put to use in this field (that is, for “plain people”).

The researcher will begin by choosing the appropriate information sources for the engagement⁸ and then move to use both specialized tools for specific sources of information and general tools like Maltego. Maltego can also be used for evidence handling, which is useful due to the large amount of data a pen-test inadvertently generates. An especially essential feature to Maltego is the ability to further search with the collected information on OSINT-sources.

Many, smaller programs exist to solve specific parts of the intelligence gathering⁹. Characteristic to these programs are that their use cases are often quite limited or with a very specific aim. Many of them are primarily concerned with foot-printing network (traffic), servers, domains and/or protection mechanisms (either passively or actively), which is a task easily done through querying open sources or the servers/domains in question.

For active fingerprinting, tools like e.g. `nmap`, `nessus`, `masscan`, `fierce`, `firewalk`, `parsero`, `smbenum` and `snmpcheck` exist. They can all perform some kind of data transaction with the targets to discover a range of properties. They are mostly scanners, which can discover information of servers and their ports, services, OS's etc. `p0f` also scans and fingerprints servers and network traffic, but do so in a semi-passive way.

For passive fingerprinting, simple tools exist to e.g. translate IP's to domain names, look-up WHOIS-information on domains, find typo-squatting domains, querying search engines, social media etc. These are the type of transforms Maltego has.

One very useful is the Shodan search engine¹⁰ which crawls and indexes IP's for meta-data about the connected devices. Users can search the database for e.g. all devices with a certain server version that are unique to uranium centrifuges. Another examples is `theHarvester`, which searches across several search engines for e-mail addresses related to a given domain.

There are a lot of these and one can readily sit down and code new ones, should some sources not be covered. There are no tools for the Danish open data sources, which is why transforms for some of these are part of the contribution of this thesis.

Tools not aimed merely at foot-printing are more rare. It is characteristic to these that it is necessary for the researcher to "add" some intelligence himself; either in the form of targeting, setup or evaluation. The tools are automating the tasks of the researcher instead of he himself wading through web pages or search engine results for documents, mail addresses or references to acquire information for the later pen-test phases.

An example hereof is `metagoofil`. It can find and extract meta-data from files, which in itself is quite simple, but added the researchers intelligence, he may be able to identify organizational

⁸Some sources are listed in http://www.pen-test-standard.org/index.php/Intelligence_Gathering#OSINT and http://www.pen-test-standard.org/index.php/Intelligence_Gathering#Covert_Gathering.

⁹A non-exhaustive list of tools could be <https://highon.coffee/blog/penetration-testing-tools-cheat-sheet/>, [59] Appendix B or the Linux distribution, dedicated to pen-testing, Kali Linux's list of tools: <http://tools.kali.org/tools-listing> (see under "Information gathering").

¹⁰<https://www.shodan.io/>

relations (e.g. who is in charge of the web site which can prove useful), confidential e-mail addresses and usernames, servers and printers or maybe even EXIF-information from pictures! Another tool `ghost-phisher`¹¹ also targets actively; it is used to emulate captive portals for e.g. company intranet sites or email phishing campaigns.

Aside from Maltego, the only other example of a tool with a wider range of use is `recon-ng`¹², which like Maltego has a number of modules for performing reconnaissance from OSINT. It can perform look-ups on search engines, resolve hostnames from IP's, find contact information, search in leak databases etc. It does however not have a case-management system and thus the information found has to be stored somewhere else.

A proper framework for collecting, combining and creating an overview of the gathered intelligence can enhance the value of the results of simple tools. As it is necessary to reuse the results from one tool and pipe it into the next one, the use case for Maltego is clear: It is both an intelligence gathering software and case-file management program (i.e. collection of intelligence on a case; not necessarily for a pen-test). I have been acquainted with a few other case-file management programs, they were hardly more than a re-skinned copy of Microsoft Office OneNote with some specialized entry types.

What Maltego does well is the graph-like representation of data and the possibility to directly search on it with provided plug-in's (called "transforms"¹³) similar to the foot-printing programs mentioned above. It is closed source, commercial software, but it is possible to develop your own transforms for it¹⁴ and range of commercial suppliers do offer both paid and free transforms (e.g. Shodan.io, VirusTotal, Kaspersky etc.) (see Figure 2.2). Developers are not obliged to publish the transforms that they develop on the Transform Hub or even use Paterva's distribution server (iTDS), so it is not possible to know the extent of non-public transforms made.

Transforms not listed on the Transform Hub by Paterva, are distributed via a *seed*, which is just an URL manually added to the Transform Hub by the user. It can provide both transforms, macro's ("machines") and configuration (e.g. custom entities or settings) to the user's local installation. All content is provided at the developer's discretion and the user cannot select part of a seed's content only.

For further on Maltego's use, please refer to Appendix A.1.

2.1.3 Threat modeling

This thesis does not aim to model threats to an organization's environment as such. It is difficult to talk about threat modeling when no specific organization has been selected and the aim of the thesis is not to develop or enhance current threat modeling frameworks.

¹¹<http://tools.kali.org/information-gathering/ghost-phisher>

¹²<http://resources.infosecinstitute.com/the-recon-ng-framework-automated-information-gathering/>

¹³See Appendix A.1 for an explanation of Maltego and the terms.

¹⁴See Appendix A for my guide to this (Paterva's documentation is a bit messy).

























+	 PATERVA CTAS <i>From Transform Hub</i> Paterva Standard Paterva Transforms FREE INSTALLED	 CaseFile Entities <i>From Transform Hub</i> Paterva Additional entities from CaseFile FREE
 SocialLinks <i>From Transform Hub</i> SocialLinks Social Networks, Search Engines, People an... PAID	 Recorded Future <i>From Transform Hub</i> Recorded Future Inc. Query Recorded Future for threat intelligenc... PAID	 Kaspersky Lab <i>From Transform Hub</i> Kaspersky Lab Query Kaspersky Threat Intelligence Data Fe... PAID
 ThreatConnect <i>From Transform Hub</i> ThreatConnect ThreatConnect Platform Transform Set PAID	 ThreatGRID <i>From Transform Hub</i> Malformity Labs Query the ThreatGRID malware platform PAID	 Shodan <i>From Transform Hub</i> Andrew MacPherson (Paterva) Query Shodan data from within Maltego! FREE
 Flashpoint <i>From Transform Hub</i> Flashpoint Search Flashpoint's dark web intelligence da... PAID	 SensePost Toolset <i>From Transform Hub</i> SensePost A set of various transforms - with regular u... FREE INSTALLED	 Intel 471 <i>From Transform Hub</i> Intel 471 Query Intel 471 for actor-centric intelligence ... PAID
 CrowdStrike Intel <i>From Transform Hub</i> CrowdStrike CrowdStrike Intelligence API Transforms PAID	 CrowdStrike Threa... <i>From Transform Hub</i> CrowdStrike CrowdStrike ThreatGraph API Transforms PAID	 VirusTotal Public API <i>From Transform Hub</i> Malformity Labs Query the VirusTotal Public API FREE
 Hyas <i>From Transform Hub</i> HYAS Inc. Reverse Whois, Phishing, Malware, and Rep... PAID	 NewsLink <i>From Transform Hub</i> Paul@Paterva Transforms for monitoring and analyzing ne... FREE	 ThreatMiner <i>From Transform Hub</i> ThreatMiner Query and pivot on data from ThreatMiner.o... FREE
 Digital Shadows <i>From Transform Hub</i> Digital Shadows Query the Digital Shadows cyber threat intel... PAID	 PassiveTotal <i>From Transform Hub</i> PassiveTotal Query PassiveTotal source and account data. FREE INSTALLED	 SocialNet <i>From Transform Hub</i> ShadowDragon Social Media Investigative Intelligence Tool PAID
 MalNet with ProofP... <i>From Transform Hub</i> ShadowDragon Maps malware intelligence. Great for IR and ... PAID	 FireEye iSIGHT Inte... <i>From Transform Hub</i> FireEye iSIGHT Intelligence Query FireEye iSIGHT Intelligence holdings. PAID	 DomainTools <i>From Transform Hub</i> DomainTools Investigate cybercrime with DomainTools hi... PAID
 Bitcoin <i>From Transform Hub</i> Paul@Paterva For visualizing the Bitcoin blockchain. FREE	 Silobreaker <i>From Transform Hub</i> Silobreaker Threat Intelligence transforms from Silobre... PAID	 ZeroFOX Transforms <i>From Transform Hub</i> ZeroFOX, Inc. Visualize ZeroFOX social media threat intelli... PAID

Figure 2.2: A screenshot of some of the transforms currently available directly in Maltego’s Transform Hub.

In this section an overview of threat modeling in general is given based on the PTES threat modeling-section¹⁵. Pointers to other methods are given in as a part of this to provide some different approaches, as the other pen-testing frameworks are virtually not considering threat modeling.

Threat modeling is essential to penetration testing as it provides the connection between the work the researcher has done and discovered and what an attacker would target. It adds a “real-life value” to the test results and the subject can prioritize mitigations better in accordance with their risk acceptance. For this thesis, the content may aid to establish common attack scenarios to tie the deliveries closer to practical examples.

Even though we say *unknown adversary* here, it will depend on the rules of engagement. The pen-tester may e.g. get credentials to act in roles of employees [45] to test resilience against misuse cases, which can arise from dissatisfied (e.g. terminated or wrongfully treated) employees [5, 29, 47, 41, 9].

The test can either include extensive communication with the subject about the possible targets

¹⁵http://www.pen-test-standard.org/index.php/Threat_Modeling

and their prioritization or the researcher might himself find several different servers during the intelligence gathering phase; the one where e.g the customer database resides is more important than the rest and emphasis during the test should hence be put here by the researcher. This would be the *primary* target. If some other programs are running here, they can be collateral, secondary targets of the test.

PTES focuses on identifying (subject) assets/processes and attackers (types and capabilities) and lists four steps of the phase from a high-level perspective¹⁶

1. Gather relevant documentation
2. Identify and categorize primary and secondary assets
3. Identify and categorize threats and threat communities
4. Map threat communities against primary and secondary assets

There are several methods available for this, as threat modeling is often an integral part of a risk analysis in order for the organization to take a complete and thorough look of their assets and asset containers (i.e. the units the assets reside on); risk analysis is treated separately in Sec. 2.2.

Two major types of approaches to threat modeling are often seen: Attack trees (system-centric/goal oriented) and misuse cases (attacker-centric)[42]. The same paper finds that attack trees are the most effective of the two for identifying threats; examples of both can be seen in Figure 2.3. The same conclusion is reached in [33] stating that “*The attacker-centric approach makes assumptions about attacker capabilities and resources.*”.

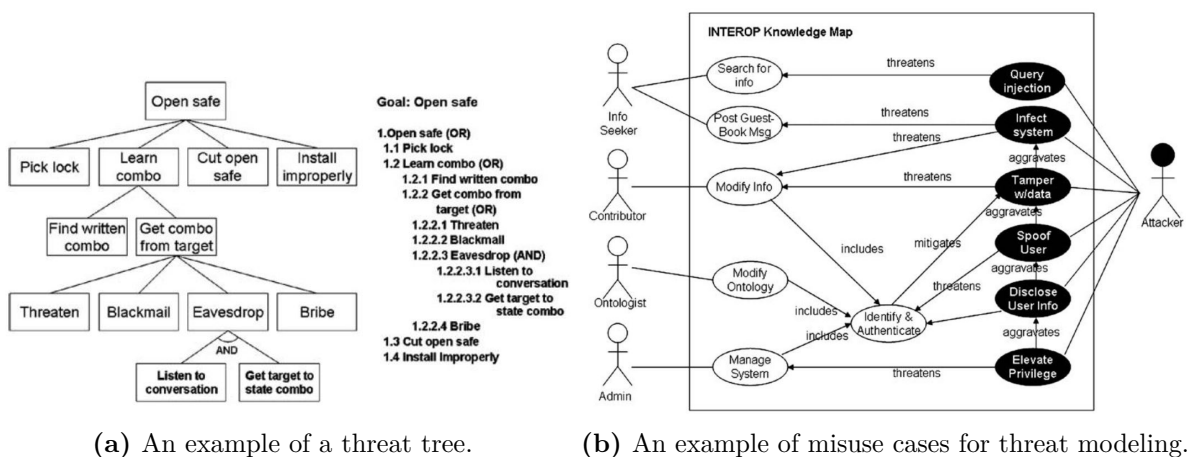


Figure 2.3: Examples of the two major approaches to threat modeling. Both taken from [42].

Several papers discuss approaches for threat modeling. Notable ones¹⁷ are [50], which originally proposed the notion and use of attack trees. [57] presents an enhancement of this “[...]for

¹⁶http://www.pen-test-standard.org/index.php/Threat_Modeling#High_level_threat_modeling_process

¹⁷I.e. referenced from other papers

expressing aggregate attack behaviors and modalities.”; [62] visits the same idea by enriching attack trees to perform advanced logic calculations to identify weakest links.

Several other papers are concerned with attack trees, as they are widely popular; EMC shows in [18] how they have used them in their software development process. If it is necessary to know more of how to use the findings in practice, [48] gives examples of applications and [36] describes how security requirements for systems can be derived from the findings. This paper states that it is ultimately always about security trade-off’s (i.e. security shall be seen in context of the risk acceptance), which is also a theme in [21], extending the i^* framework¹⁸ to enhance the analysis of the trade-off. [31] presents similar work, also based on i^* , to derive security and privacy requirements from the threat modeling.

A more formal approach to attack trees is presented in [1], where an IDS is modeled using attack trees. The approach of [33] does not have a direct practical value as presented in the paper, but may be of academic interest; it contains a great deal of references.

For misuse cases, [52] is one of the papers originally describing the technique; here by extending regular use cases with “negative” uses cases (i.e. uses cases intended to not fulfill); [42] gives references to the others. Off the same idea, [40] uses the same notion of a negative use case to create attack trees with negative (“soft”-)goals, which can offer a differentiated view of the attacker goals.

To get a view on the practical use with some insight, great examples are the aforementioned [18] and also [51], which presents Microsoft’s system-centric, practical approach to threat modeling. It may be valuable to the analyst to look into current attacks methods in the same way it is done in Section 2.5 and 2.4 on current attacks and social engineering techniques respectively. Focus in those sections are not on overall attack methods, but only within social engineering.

It should be noted, as with risk assessments in general (see also Sec. 2.2), also the threat modeling has to be maintained to continuously return value to the organization for the time invested in the work. By now, supposedly many commercial tools exist to automate this to some extent. [26] presents an early, theoretic approach for this.

2.1.4 Vulnerability analysis

The goal of the vulnerability analysis is to discover and validate flaws identified previously, which can be used by an attacker to compromise the system to achieve his goals¹⁹. These are collected, validated and organized s.t. they can be leveraged as a part of the exploitation phase of the pen-test.

Parts of the intelligence gathering phase (Sec. 2.1.2) are repeated in this phase as a way to obtain an overview. If only a passive or semi-passive reconnaissance were made, a more active approach may be necessary for this phase.

¹⁸Used for requirements engineering; https://en.wikipedia.org/wiki/I*

¹⁹Attacker goals are a part of most threat model and risk analysis-frameworks as mentioned in Sec. 2.1.3

The collection of the vulnerabilities does not differ much from what was explained in Sec. 2.1.2. Validation may already have been part of the previous work, but will otherwise demand further testing of the reported output of the tools already run.

The grouping of the identified vulnerabilities is a key outcome of this phase as it improves the value of the results by a better overview and understanding of what is found. Multiple vulnerabilities on the same piece of software/service are grouped and will thus not skew results.

PTES suggests utilizing two methods (not mutually exclusive): *Attack trees* (explained in Sec. 2.1.3) or aggregation by “vulnerability ID’s” (e.g. the CVE²⁰ or NVD²¹-database).

CVE-ID’s are very widely used and virtually all vulnerabilities are recorded here. The ID’s are giving to applicants at the discretion of the manufacturer or a central organ according to some guidelines²² with the aim to “[...]be comprehensive with respect to all publicly known vulnerabilities and exposures.”²³. The ID’s are great for tracking information on vulnerabilities and when used in a report, it enables the reader to look up details across many sources.

Useful for the researcher to categorize or come up with additional vulnerabilities, [37] distinguishes between *flaws* (unintended functionality from poor design or mistakes during implementation; most types of vulnerabilities found today), *features* (intended functionality misused, like built-in diagnostics or macros) or *user errors* (default/weak passwords, giving up information, installing malware, clicking on stuff).

Finally, the findings should be researched and validated against other sources [45, 56]. This is to ensure that the findings are actually valid (e.g. the vulnerability is mitigated elsewhere or is a false positive, by e.g. not being applicable to the OS).

2.1.4.1 Vulnerability scans as independent reports

A vulnerability scans in itself can also be of value to an organization. This is a requirement of e.g. the ISMS-system in DS/ISO 27000-series, specifically found in DS/ISO 27001 [15] control A.12.6.1 (“Management of technical vulnerabilities”) and control A.18.2.3 (“Technical compliance review”), which can be measured by construct DS/ISO 27004 [17] B.29 (“Pen-test and vulnerability assessment”), as well as the PCI DSS (see [45] Sec. 2.1).

As mentioned in Sec. 2.1 it is as a standalone report; a smaller, less resource-demanding process aiming to “*Identify, rank, and report vulnerabilities that, if exploited, may result in an intentional or unintentional compromise of a system.*” [45] instead of aiming to penetrate the network (which may require only a single vulnerability) [59]. The report will run automatically scanning for open ports and indicators of vulnerabilities as pre-recorded by e.g. a vendor database.

The pen-tester can use it to continue his test as described, but even with only this data in hand,

²⁰<https://cve.mitre.org/>

²¹<https://nvd.nist.gov/>

²²https://cve.mitre.org/cve/editorial_policies/index.html

²³https://cve.mitre.org/about/faqs.html#cve_list_contain_all_vulnerabilities

the organization can already know where to e.g. update software or limit outside access. Coupled with the comparatively quick survey time (and thus lower cost?), it may be utilized more often than the pen-test for hardening the organization's information systems.

The pen-tester should make sure to include the results of his vulnerability scan in the final report to let the client organization gain the same insight [59].

Examples of reports of vulnerability scans can be found in Appendix B; their setup/content are further discussed in Sec. 2.1.6.

2.1.5 Exploitation

Exploitation is not a subject of this thesis; this is a brief overview of what the phase can encompass.

In this phase, the vulnerabilities are exploited to gain access to the organization's environment. PTES has split this phase into two parts concerned with planning and gaining access (as easy as possible, [56] suggests) to systems with the greatest impact to the organization's business and "[...]to determine the value of the machine compromised and to maintain control of the machine for later use."²⁴ [56] respectively.

[59, 45] does consider this as one, but it makes sense to first plan, then execute²⁵.

The pen-tester should first consider and verify attack strategies to avoid countermeasures installed like IDS/IPS, antivirus, firewalls or humans; a non-exhaustive list of strategies are given in [56, 59, 45, 49], but none of them has specific pointers.

For this phase, the pen-tester will thus turn to practical handbooks on the subject like [30, 61] of which there are plenty. It will be a good idea to get recommendations for specific handbooks if possible, as there may be a difference in quality.

Second the pen-tester should execute the plan. [56] contains guidelines on analyzing the infrastructure to which access has been gained, finding and extracting information from them, creating persistence on the systems and leveraging this to further compromise other systems on the network. Finally, a note on clean-up of the work performed is given.

For this specific phase, the researcher can employ a framework like the **Social Engineer Toolkit**²⁶ (SET). This framework allows for the sending of e-mails falsely seeming to come from internal, fake Java-applets, Metasploit-modules, SMS spoofing and a lot more.

²⁴... which can be a re-iteration of previous phases to use the exploited system to gain access to other parts of the organization's systems.

²⁵The classic "plan, do, check, act"-approach. See e.g. <http://asq.org/learn-about-quality/project-planning-tools/overview/pdca-cycle.html>

²⁶<https://www.social-engineer.org/framework/se-tools/computer-based/social-engineer-toolkit-set/>

2.1.6 Reporting

The goal of the report on the pen-test is to provide the client organization with the insight gained in the test and do so in a easily digestible format/overview, that does not require the pen-tester's expertise to understand. To provide an "easily digestible format with the insight gained", does not mean to state everything at once, but instead to prioritize and section the report s.t. most severe vulnerabilities are presented (based on the pen-tester's insight) for the client to remedy first.

"A report should be generated that identifies system, network, and organizational vulnerabilities and their recommended mitigation actions."[49] and it *"[...] should be structured in a way to clearly communicate what was tested, how it was tested, and the results of the testing."*[45]. It is however highly dependable on what the client planned the test to be used for.

[56] provides a good template²⁷ for making a report on the results of the pen-test. It has the necessary categories to get a full report made and provides suggestions for the text (which the others do not).

The template is split into two parts: An executive summary and the technical report. The template lacks formal sections, which are prevalent in the templates of [45, 49]. These are included in the example structure below. [59] are some-what brief in its description of the reporting, but follows the same pattern.

- Executive summary
 1. **Background** and goals of the test and details of the agreed setup around the test.
 2. A brief of the **overall effectiveness and achieved work** in relation to the pre-conditions (including any limitations).
 3. A **risk profile** based on the findings using e.g. colors and absolute values (see Sec. 2.2 for more on this).
 4. **General findings** *"[...]in a basic and statistical format"*[56] using e.g. graphs and tables of summaries.
 5. **Recommendation summary** and **strategic roadmap** to help the organization understand where to put in the efforts. The pen-tester will use the pre-defined objectives and understanding of the business impact of the systems to prioritize mitigations for the client.
- Technical report: A report of the (ranked) findings with references and a conclusion containing recommendations to mitigate and secure the organization in the future as in the last part of the executive summary (in more technical terms); this includes suggestions and techniques to resolve them.

Concludes with detailed listings all information gathered during penetration testing and all vulnerabilities found.

²⁷Judging from my own, practical work experience of 3 years on making executive reports of vulnerability scans and web application tests at my student job.

This is an exhaustive list of the usual contents of a pen-test report. The actual organization of the contents, the wording, the scales and the charts used will of course vary across different researchers depending on the exposure they have had to different reports throughout their career and their personal preferences. For the executive summary and especially the statistics, all of the considered sources are lacking suggestions on how to visually present the contents.

In Appendix B.1 a real-world example of a vulnerability scan from the vendor Qualys is shown. It is a scan run against three IP's of a client by Dubex A/S Security Analytics Center (SAC) and gives an idea of how a professional report look. This is obviously not a fully comprehensive example, but good enough to have been sold to a lot of enterprise customers.

Additionally, in Appendix B.2 a generic example report (also from Dubex SAC) is shown. This is an “executive” summary of a Qualys vulnerability scan (not related to the scan report in app. B.1) and shows how the most important parts can be highlighted (a short executive summary, statistical overview of hosts and their degree of vulnerability and the most important vulnerabilities found with mitigations).

These real-world examples can be used to get an idea of how one might structure a vulnerability report, but emphasis should be made that the pen-tester may have different preferences, knowledge of better ways to structure his specific statistics or findings or other goals of the specific engagement.

2.2 Risk analysis

As mentioned in Sec. 2.1.3, threat modeling is often an integral part of a *risk analysis* or *risk assessments*²⁸. Several models and standards exist; this section covers the most reputable ones including those part of international guidelines. Methods for performing a risk analysis is not directly concerned with this thesis, but knowledge of it provides a baseline for understanding the processes the researcher or organization might follow to map the threat landscape and where this thesis' products eventually will fit in. It can also be used for a reader to select an appropriate risk analysis framework.

2.2.1 A general view: DS/ISO 27000-series

[13, 54] both presents standardized guidelines for a general approach of risk assessment. For standardization frameworks like these, “A *single risk model* [...] *cannot meet the diverse needs of the organizations* [...] *that rely on [standards]*” [54] (i.e. DS/ISO 27000 and NIST 800-30) and thus the standards presents best practices and guides the organizations towards understanding what risks are important to them.

We naturally use the DS/ISO 27000-series here (see also Sec. 2.3.1), as it is a standard directly applicable to Danish organizations by EU-law. [13] lists the following activities of a risk assessment:

²⁸The use of the terms vary and often a risk analysis is only a part of an overall risk assessment.

- Risk identification
- Risk analysis
- Risk evaluation

The **risk identification** is the activity to identify assets, threats to these, existing controls (procedures or mitigation plans), identification of vulnerabilities and the consequences they may have on the assets (measured against the classic CIA-properties of *confidentiality, integrity and availability*²⁹, often includes non-repudiation as well).

The **risk analysis** activity details two general types of methodologies [13]:

Qualitative risk analysis “[...]uses a scale [...] to describe the magnitude of potential consequences (e.g. Low, Medium and High) and the likelihood that those consequences will occur.”. This approach can be easier to interpret to all personal, but is dependent on a well-chosen scale. It can be used as an initial screening and where resources or numerical data are insufficient to perform a quantitative analysis.

Quantitative risk analysis “[...]uses a scale with numerical values (rather than descriptive scales [...]) for both consequences and likelihood, using data from a variety of sources.”. This approach is dependent on both the model to use well-chosen values and that the sources used are complete. If not, the lack of valid data may hide new risks and weaknesses from the results.

It is advised that the chosen methodology aligns with the chosen risk evaluation criteria such that the necessary insight and value is gained in the end. In practice it is often found that the risk analysis models combine both of the above types of methodologies to some extent.

The **risk evaluation** is the activity of comparing the risks and their connected values to the risk evaluation criteria established for the organization. Here, risk are prioritized according to the information security properties of the organization (e.g. confidentiality may not be relevant and should thus be ranked lowest or even removed) and the assets’ importance. From this, decisions on mitigation of risks can be made³⁰; this process is further described in [13] (page 20 and onward), but out of scope of this report.

For mature organizations, the risk analysis/risk assessment is an ongoing process, where the analysis is regularly maintained [6, 13, 54].

For context, the risk assessment as described here, is a part of the overall information security risk management process as described in ISO 31000 and shown in Figure 2.4. The take of [54] on the same process is depicted in Figure 2.5, with a close-up of the risk assessment process in itself in Figure 2.6. Despite different wording, the processes are quite identical.

²⁹A short explanation can be found here: <http://whatis.techtarget.com/definition/Confidentiality->

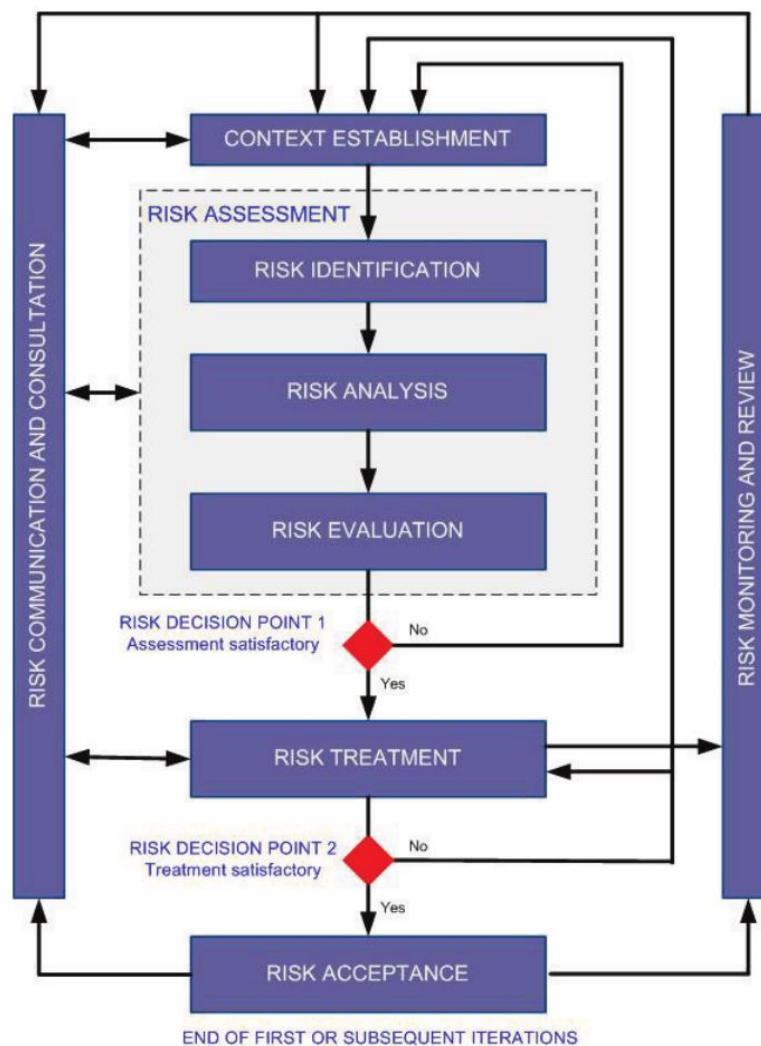


Figure 2.4: The full risk management process of ISO 31000 as applied in the DS/ISO 27000-series. From [13].

2.2.2 Specific methodologies

[58] presents a (qualitative) framework for comparing risk analysis models. As examples, they compare OCTAVE (not OCTAVE Allegro, which is a later, revised model of OCTAVE but for all organizational sizes[6]), CORAS [32], ISRAM³¹, CORA³² and *IS Risk Analysis Based on a Business Model* by KAIST³³.

OCTAVE and CORAS are characterized as qualitative models, while the others are quantitative [58]. They have chosen these, as they are well-documented [58]. I include OCTAVE Allegro too,

integrity-and-availability-CIA (accessed 2017-04-30).

³⁰The mitigation process in [13] is called *risk treatment*. A full wordlist of the DS/ISO 27000-series on information security management systems is found in [14].

³¹<https://www.sciencedirect.com/science/article/pii/S0167404804001890?np=y>

³²Not very used at all and difficult to find anything about, as the maintaining organization does not exist any longer. Short description here: <http://www.blacksheepnetworks.com/security/resources/encyclopedia/products/prod131.htm>.

³³<http://koasas.kaist.ac.kr/handle/10203/3686>

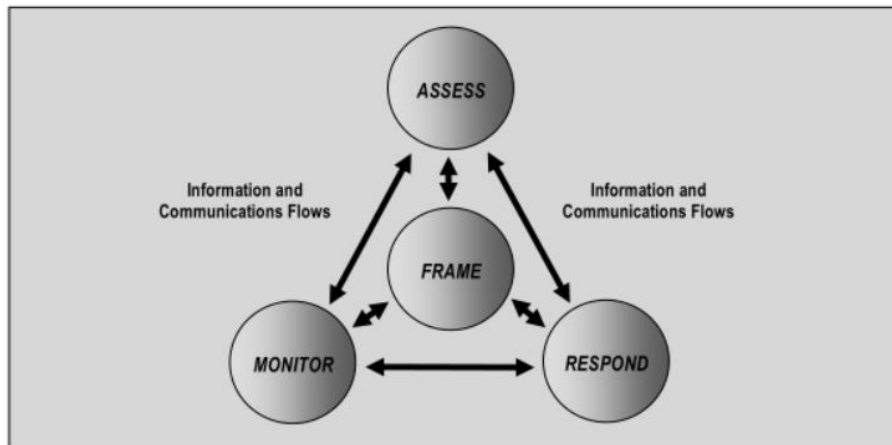


Figure 2.5: The risk management process as depicted in NIST 800-30 [54].

which is a successor to OCTAVE and thus quite similar, but more versatile [6].

In the following we use the criteria used in [58] to describe what differences risk analysis methodologies have.

In general, but depending on the specific model utilized, the organization identifies their assets and then formulates specific requirements of the security of the organization as a whole (most likely based on current standards). The threat modeling, which will tell something about the *probability* of attack scenarios on the assets, is combined with the *impact* some attack scenario will have on the organization to return a *risk score*. This is typically done using a risk matrix (see an example in Fig. 2.7), where values (discrete or continuous cf. Sec. 2.2.1) of probability and impact are assigned to the scenarios for prioritization of the results.

Combined with an assessment of the mitigations in place and the risk acceptance, the organization can then take an informed stance on the threats against them and where to put in an effort to improve the results cf. the above.

It is noteworthy how the identification of assets and subsequently the attack scenarios against them differ from model to model; in [58] this is characterized as “*Whether risk analysis is done on single assets or groups of assets*”, whilst it does not look on how the attack scenarios themselves are defined, as it is not a property of all the five examples used.

One approach is to perform a pen-test-like scan much like PTES, where tools are used to identify attack scenarios against either pre-defined assets or where an intelligence gathering-phase (as described in Sec. 2.1.2) are to discover them.

This is opposite to “asset/asset container”-driven models like [6, 32], where “*[i]nstead of running vulnerability tools and using the results to seed threat identification, in OCTAVE Allegro users map an information asset to all of the containers in which it is stored, transported, or processed and consider threats to each of those containers. There is still a technology view, but it is not impeded by the execution of cumbersome tools that require specialized knowledge and resources.*” [6]. As IT security has become an increasingly complicated field that system owners has to prioritize among many other specialized skills, such models have won popularity, as they are

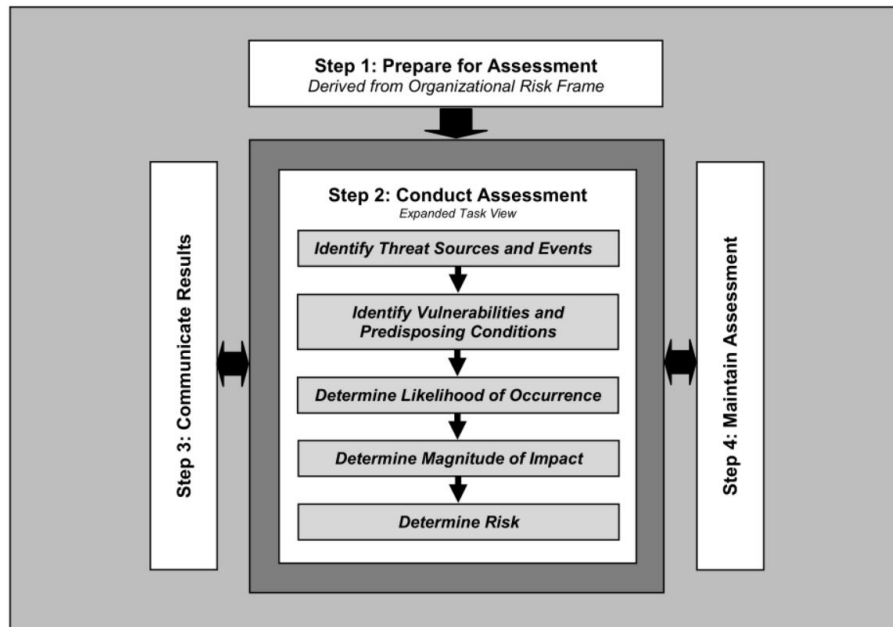


Figure 2.6: The risk assessment process (part of the risk management process, see Fig. 2.5) as depicted in NIST 800-30 [54].

easier to use and does not demand external consultancy aid.

[58] uses “*The main formulae used*” to cover this question, as it relates to exactly the “easeness” of use.

This in turn is connected to the property of interval vs. external aid from the criteria “*The people involved in the risk analysis*”; All of the mentioned methodologies, except for CORA, are internally focused, which supports the above claim of its popularity.

Within these two extremes, we also see differences in the way the attacker is analyzed. Some models use an approach like PTES, where one try to find all possible attack vectors using prior knowledge, research and automated tools; attack trees as described in Sec. 2.1.3 are an example hereof, where a methodically “top-to-down”-approach is used beginning with the attacker type. Other models like OCTAVE Allegro [6] and Microsofts framework [34]³⁴ applies an “attacker goal-oriented” approach applied from inside and out. Here, the models asks the user to methodically start by establishing what types of goal an attacker might have against each asset/asset container; this is a great approach for “regular” teams of developers, where security knowledge is not prevalent [51].

The other criteria used in [58] are whether results are comparable across multiple applications of the method (for OCTAVE (Allegro), CORAS and ISRAM they are not) and the amount of preparation done before the actual risk analysis. OCTAVE (Allegro) and ISRAM does a great deal beforehand, CORAS a medium amount, while CORA and IS a minimal amount.

³⁴The acronyms STRIDE and DREAD used for threat classification respectively calculation of the risks (quantitatively) are a part of this.

		Impact →				
		Negligible	Minor	Moderate	Significant	Severe
Likelihood ↑	Very Likely	Low Med	Medium	Med Hi	High	High
	Likely	Low	Low Med	Medium	Med Hi	High
	Possible	Low	Low Med	Medium	Med Hi	Med Hi
	Unlikely	Low	Low Med	Low Med	Medium	Med Hi
	Very Unlikely	Low	Low	Low Med	Medium	Medium

Figure 2.7: Example of a risk matrix identical to the one used in e.g. CORAS [32]. Taken from <https://causalcapital.blogspot.dk/2015/08/making-risk-matrix-useful.html>.

For further on this, we refer to [58] directly.

2.3 Standards and guidelines

This section is concerned with national (Danish) regulation, standards and guidelines intended to (among others) hinder attacks enabled by OSINT-data³⁵ and hinder generation of this data by the organization.

The primary aim of standards are to give controls indicating what to do, but not a policy directly adoptable for the organization. Instead controls are “rules” which are to be complied to, so the implementing group will have to find concrete advice in other sources like guidance from official organs, which is also considered in this section.

The primary aim of a legislation is to direct procedures and actions of the organization in general, while actual implementation advice only comes in the form of concrete guidelines (as we see it currently with the guidance interpreting GDPR in Danish context³⁶ Both Danish regulation, standards and guidelines are presented in this section.

It is valuable to survey current regulation in this section, as it is obligatory to an organization to act under these sets of rules and at all times evaluate their actions against them.

It is valuable to include controls of a standard or guidelines in this project, as compliance with some standard is an easily communicable *key performance indicator* (KPI) to e.g. its external stakeholders – be it customers, suppliers, NGO’s or government entities. As opposed to current legislation, standards and guidelines are not required for the organization to adhere to, but can serve as guidance on operations.

³⁵I.e. data generated from the organization’s daily operations and found on public sources like the organization’s web site, 3rd party websites/search engines or in government registers. Some sources also use the term “meta-data”.

³⁶See more in Sec. 2.3.8 or at <http://justitsministeriet.dk/nyt-og-presse/pressemeldelser/2017/nye-regler-styrker-beskyttelsen-af-persondata-i-europa>.

An increasing number of organizations however adhere (or plan to) the DS/ISO 27000-series³⁷. The series is currently the primary way for organizations to demonstrate that IT security is considered and that the organization's practices are secure, though that is not implied just by adopting the standard – the organization needs to be audited and certified according to the standard.

We can thus use the standards and guidelines to have a solid base for relating the results of the generated report.

During the research of relevant regulation, standards and guidelines, it is found that while the maturity for security frameworks for actual data assets and their surroundings are great, the material lacks as soon as we move towards “outbound information sharing” and controlling the organization's generation of OSINT-data and the handling of it [25]. Regulations and standards speak about a general need of awareness training etc., but actual guidance on what/when/how to handle outbound information sharing is hard found.

Obviously there are some inherent difficulties in describing a standard for highly dynamic world or a non-deterministic entity³⁸ as opposed to a mostly deterministic IT-system: The human mind is biased and will deviate from normal procedures to in some cases (which is exactly what the social engineer will exploit; see Sec. 2.4 for a brief view on human mind and social engineering techniques and Sec. 2.5 for current attacks exploiting this.).

With the above in mind, relevant controls for OSINT-data of the surveyed material are included below; and even some that may not be directly apply, but can be extended to cover OSINT-data. This sections also lists guidelines discussing awareness or what can be shared online on e.g. social media, which is also considered an OSINT-data source. We skip the content of Danish laws on the subject due to lack of concrete advice (see Sec. 2.3.7).

A selection between the content of the standards and guidelines has been made. Focus is data generated from e.g. the organization's daily operations about infrastructure, employees, assets etc. that can enable an attacker to gain access to the organization's “real” assets, e.g a social engineer (see Sec. 2.5 for current examples of attacks). This is what [25] names “outbound information sharing”. Some of the omitted controls are relevant in protecting against this, but are so general that they are not valuable to include, because they cannot be referenced at a later point to be coupled with a the finding of a specific piece of information.

An example: A control directs to partition the network into different zones and to implement a web filter. This may aid to hinder access to a site, where OSINT-data can be shared, or hinder an attacker to gain a foothold. However this controls will not hinder a user to share job descriptions on Facebook or working procedures on Reddit.

Contrary, a control directing the users to have awareness training on social engineering attack

³⁷See https://www.iso.org/files/live/sites/isoorg/files/standards/conformity_assessment/certification/doc/survey_executive-summary.pdf and <https://www.itgovernanceusa.com/blog/iso-reports-a-78-increase-in-us-based-iso-27001-certifications/>

³⁸I.e. the human mind, which since the mid 70's has been perceived as acting and thinking with systematic errors.

types and regular exercises can directly hinder the above, as the user is now aware of the risk of doing so and are reminded regularly hereof.

If the sources behind are referenced by the reader, more thorough guidance can be found. Indications have been made on sources which can be particularly helpful to look further into.

2.3.1 DS/ISO/IEC 27000-series

The series are made by work groups under the International Organization for Standardization (ISO) and the European Committee for Standardization (IEC) and adopted as a whole by the Danish Standards Foundation (DS). It is a standard for an *Information Security Management System* (ISMS). The relationships between the standards in the “ISMS family” can be seen in Figure 2.8.

The DS/ISO 27000-series are not necessarily used in an organization as a direct set of rules, but can also act as baseline for a set of guidelines, with which the organization controls cyber security (all aspects or elements hereof).

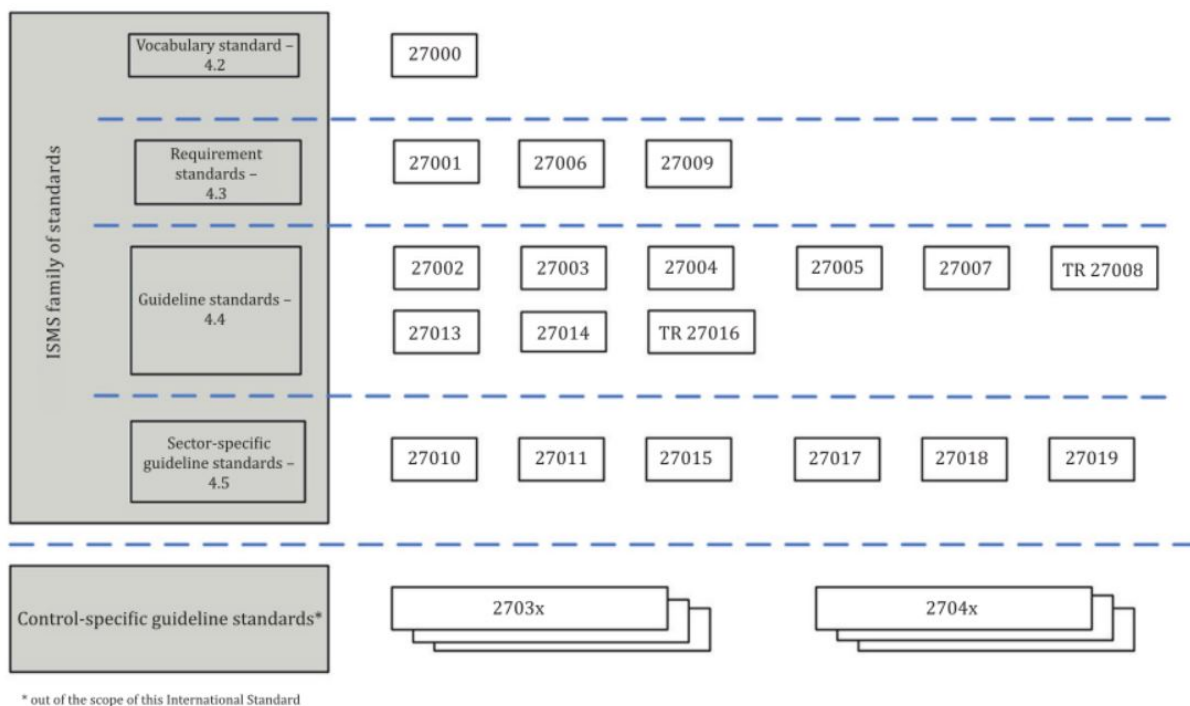


Figure 2.8: Relationships of the ISMS family of standards. From [14].

In DS/ISO 27000 [14] an overview and vocabulary for the standards are given.

DS/ISO 27001 [15] describes the requirements of an ISMS, which is what an organization can certify its conformity to. These are provided in 10 clauses with subclauses. In Annex A a comprehensive list of controls with objectives are given; these are to be used as a part of a specific subclause (6.1.3 b)) to perform information security risk treatment within the organization. In this way, they guide the organization on how to put the intentions into concrete, implementable controls.

DS/ISO 27006 and DS/ISO 27009 (in the same category as 27001) in Fig. 2.8, are requirements for certification bodies respectively requirements for sector specific implementations of the ISMS. The rest of the standards provide guidance for a general ISMS-process as well as sector-specific guidance.

DS/ISO 27002 [16] is also reviewed; it contains the specific “code of practice for information security controls” which helps an organization to adhere to the requirements (the concrete rule-set) found in DS/ISO 27001.

DS/ISO 27004 [17] provides guidelines to assist an organization to measure the performance and effectiveness of an ISMS as required by DS/ISO 27001, clause 9.1. The guidelines are directly mapped to a subset of the controls of DS/ISO 27001 and can be adopted in the organization.

DS/ISO 27005 [13] is the standard for risk analysis; it is referenced in Sec. 2.2.1 on risk analysis.

2.3.1.1 Controls from DS/ISO 27001 to hinder unintended leaks of organization data

The following section provides a list from DS/ISO 27001 Annex A on controls for information security risk treatment, and lists the content of each control as described in DS/ISO 27002; they can be seen in Table 2.1. We include only controls that are deemed relevant to this thesis. Each control has been reviewed for relevance to this thesis’ subject of elevating control with data flows to OSINT-sources. Some of the standard’s controls can be proven to be directly violated with concrete findings in a pen-test, while others are more “meta” (i.e. they must have been violated, since data XYZ where found). The rest are left out.

The requirements of DS/ISO 27001 are meant to be generic, applicable to all organizations and to be “checked off” by an auditor, whereas the controls of Annex A are concrete. Hence it is more relevant to look into the actual controls offered in Annex A for risk treatment, as it provides tangible understanding of what kind of controls organizations should or will have implemented and adhere to. In this way we can make the results of a test of the organization more directly recognizable to them and their daily operations.

<i>Control</i>	<i>Title</i>	<i>Content of control</i>
A.7 Human resource security		
A.7.2 During employment		
A.7.2.1	Management responsibilities	Management shall require all employees and contractors to apply information security in accordance with the established policies and procedures of the organization
A.7.2.2	Information security awareness, education and training	All employees of the organization and, where relevant, contractors shall receive appropriate awareness education and training and regular updates in organizational policies and procedures, as relevant for their job function.
A.8 Asset management		
A.8.1 Responsibility for assets		

A.8.1.3	Acceptable use of assets	Rules for the acceptable use of information and of assets associated with information and information processing facilities shall be identified, documented and implemented
A.8.2 Information classification		
A.8.2.3	Handling of assets	Procedures for handling assets shall be developed and implemented in accordance with the information classification scheme adopted by the organization. ³⁹
A.13 Communications security		
A.13.2 Information transfer		
A.13.2.1	Information transfer policies and procedures	Formal transfer policies, procedures and controls shall be in place to protect the transfer of information through the use of all types of communication facilities.
A.13.2.3	Electronic messaging	Information involved in electronic messaging shall be appropriately protected.
A.15 Supplier relationships		
A.15.1 Information security in supplier relationships		
A.15.1.3	Information and communication technology supply chain	Agreements with suppliers shall include requirements to address the information security risks associated with information and communications technology services and product supply chain. ⁴⁰
A.18 Compliance		
A.18.1 Compliance with legal and contractual requirements		
A.18.1.3	Protection of records	Records shall be protected from loss, destruction, falsification, unauthorized access and unauthorized release, in accordance with legislative, regulatory, contractual and business requirements.
A.18.1.4	Privacy and protection of personally identifiable information	Privacy and protection of personally identifiable information shall be ensured as required in relevant legislation and regulation where applicable.
A.18.2 Information security reviews		
A.18.2.2	Compliance with security policies and standards	Managers shall regularly review the compliance of information processing and procedures within their area of responsibility with the appropriate security policies, standards and any other security requirements.

Table 2.1: Policies relevant to this thesis from DS/ISO 27001. © Danish Standards Foundation

As can be seen, the controls are very general. Many of them are technical, which is often also what is thought about, when mentioning information security, but the security is ineffective if it

³⁹A.8.2.1 and A.8.2.2 are closely related, but A.8.2.3 is the primary control.

⁴⁰A.15.1.1 and A.15.1.2 are closely related, but A.15.1.3 is the primary control.

is not supported by management and procedures [14]. Thus in DS/ISO 27002 further guidance with recommendations⁴¹ on how to achieve the objective of the controls are found. Of special interest is the guidance for A.7.2.1, A.7.2.2, A.13.2.1 and A.13.2.3.

The guidelines in full length is too long to list here; coming from copyrighted material, the pdf is protected of copy-paste operations and I may also infringe on copyright. Hence the following paragraphs will shortly list the noteworthy parts of the four controls. The interested reader are directed to [16] for further.

2.3.1.1.1 A.7.2.1 Management responsibilities The standard notes the importance of in part briefing employees of their security roles and responsibilities (prior to being granted access to confidential information) and in part adhering the employees to the security policies as part of their contractual working agreement. It is the management's responsibility to ensure this. Awareness, motivation, development of employee skills, etc. are also an integral part of this. If management fail to do this, the organization may suffer or be liable for considerable damage.

2.3.1.1.2 A.7.2.2 Information security awareness, education and training The implementation guidelines for this control addresses the aim, establishment, general content and use of the awareness programme of the organization. The goal of this standard's programme is to make employees aware of their *responsibilities* and how these should be conducted. This is different from the advice from the sources in the following sections, which emphasizes knowledge of attack scenarios.

The programme can be constructed from lessons learned from previous incidents and should of course be in accordance with the rest of the organization's policies. When building it, it is recommended to focus not only on *what* and *how*, but also *why*. This can increase vigilance by employees, as they understand the positive and negative impact their actions may have. In relation to this, the programme should contain basic procedures (e.g. reporting) and controls, to show the employees how they can protect and react to threats.

The training should be performed at employment and regularly afterwards and complemented by tests. The training should be presented using different methods and media to adhere to the broadest audience possible. It should be relevant to the employees to which it will be presented – and also cover contractors as necessary.

It is not mentioned here, but the organization can benefit from looking into other organization's examples of awareness programmes and adjust and use them according to their needs and the standard, as no concrete examples are given here.

2.3.1.1.3 A.13.2.1 Information transfer policies and procedures This implementation guides what to take into account when forming procedures for information transfer through “communication facilities” (everything else than face-to-face transactions, it is understood). Of

⁴¹Not requirements; the distinction is standardized for the ISMS and can be found in Annex A of [14].

special interest is advice for transfer of the data, which is not the common assets considered (as described in the introduction, Sec. 1):

Employees should take care not to reveal confidential information – not only at work, but at any time. It also advice of highly specific threats like leaving messages on voice answering or facsimile machines, misdialing, making a typo in an email address or automatic mail forwarding.

Apart from this, the advice in this Section is technical e.g. recommending to check the policies of all communication media used (examples hereof is given in the document) and employing best-practices of cryptography use.

2.3.1.1.4 A.13.2.3 Electronic messaging This recommendation is concerned with the protection of information in transfer. It notes the importance of directing/addressing the information correctly, which is a big issue in regards to outward shared information and OSINT-enabled attacks (see Sec. 2.5 for examples of this); employees will need to strictly be sure that the destined receiver of the communication is who he claims to be.

The rest of the advice in this Section is purely technical and thus out-of-scope for this thesis.

2.3.1.2 DS/ISO 27004 examples of measurements for controls from DS/ISO 27001

DS/ISO 27004 Annex B [17] gives examples of specific ways to measure some of the controls of DS/ISO 27001 Annex A [15]. Annex B in [17] also offers examples of measurement of the requirements in DS/ISO 27001.

This section lists a couple of these to give an immediate picture of how the ones responsible in the organization can measure their compliance with the controls.

2.3.1.2.1 B.14 ISMS awareness campaigns effectiveness This construct relates to control A.7.2.2 of DS/ISO 27001.

Information descriptor	Meaning or purpose
Measure ID	Organization-defined.
Information need	To measure if employees have understood content of awareness campaign.
Measure	Percentage of employees passing a knowledge test before and after ISMS awareness campaign.
Formula/scoring	Choose a number of employees who were targeted by an awareness campaign and let them fill out a short knowledge test about topics of the awareness campaign. Percentage of people passed the test is used for scoring.
Target	Green: 90 – 100% of people passed the test, Orange: 60 – 90% of people passed the test, Red: < 60% of people passed the test.

Implementation evidence	Awareness campaign documents/information provided to employees; list of employees who followed awareness campaign; knowledge tests.
Frequency	Collect: One month after awareness campaign. Report: For each collection.
Responsible parties	Information owner: HR. Information collector: HR. Measurement client: Information security manager.
Data source	Employee database, awareness campaign information, knowledge test results.
Reporting format	Pie chart for representing percentage of staff members passed the test situation and line chart for evolution representation if extra training has been organized for a specific topic.

Table 2.2: Example of how measurement of an ISMS awareness campaign can be performed and reported. From [17] © Danish Standards Foundation

2.3.1.2.2 B.31 Security in third party agreements - A This construct relates to control A.15.1.2 of DS/ISO 27001.

Information descriptor	Meaning or purpose
Measure ID	Organization-defined.
Information need	To evaluate the degree to which security is addressed in third party agreements.
Measure	Average percent of relevant security requirements addressed in third party agreements.
Formula/scoring	$[\text{Sum of (for each agreement (number of required requirements - number of addressed requirements))} / \text{number of agreements}] * 100$
Target	100%
Implementation evidence	Supplier database, supplier agreement records.
Frequency	Collect: Quarterly. Report: Semi-annually.
Responsible parties	Information owner: Contract office. Information collector: Security staff. Measurement client: Security manager, business managers.
Data source	Supplier database, supplier agreement records.
Reporting format	Line chart depicting a trend over multiple reporting periods; short summary of findings and possible management actions.

Table 2.3: Example of how measurement of the security in third party agreements in the organization can be performed and reported. From [17] © Danish Standards Foundation

Apart from the above examples, DS/ISO 27004 Annex B offers examples to measure controls like e.g. change management, log reviewing, device configuration, pen-tests and incidents cost.

Those are not related to the relevant controls of Section 2.3.1.1.

2.3.2 Centre for Cyber Security (part of Danish Defence Intelligence Service)

“‘Centre for Cyber Security’ is a sector of the ‘Danish Defence Intelligence Service’. CFCS is a national information and communications technology (ICT) security authority. It is an independent authority governed by separate legislation. As the overall national ICT security authority, the centre has three primary responsibilities:”⁴²

- Contribute to protect Denmark against cyber threats
- Assist in securing a solid and robust ICT critical infrastructure in Denmark
- Warn of, protect against and counter cyber attacks

Part of this contribution is materialized through guides published on current IT security subjects relevant for Danish organizations.

2.3.2.1 “Cyberforsvar der virker”

In the second version of the guidance “*Cyberforsvar der virker*” (“Cyber defense that works”) [20] the Danish Agency of Digitisation and Centre for Cyber Security under the DDIS describes a concrete, prioritized plan for how government and private organizations can reduce the risk of cyber attacks and handle the worst consequences when an attack hits. They note the importance of management support for the changes in policies and culture to succeed.

The guidance is made in seven steps including four “basic” security measures, *Top 4*, which are being adhered to/followed by a large number of countries and government cyber security centres [20]. The guidance claims that by following all the advice in it, up to 80% of cyber attacks can be avoided.

In title headings, the seven steps of the guidance are:

1. Top-management support
 - Understand the threat, support the defense and delegate daily responsibility.
 - Complete an overall IT-security risk assessment
2. The right technical competences
 - Make sure that the organization possesses the right technical competences or have access to them.
3. The basic security measures
 - Implement security measure to secure high risk targets/-assets.

⁴²Taken from <https://fe-ddis.dk/eng/About-DDIS/Pages/Organization.aspx>

- Extend this to the remaining targets/assets at risk afterwards.
4. Awareness, awareness, awareness
 - Introduce the security policy to new hires.
 - Continuously inform about the cyber threat.
 5. A reactive capacity
 - Start small and prioritize high risk-targets/-assets.
 - Establish relevant reactive competences.
 6. Continuous security technical investigations
 - Continuously test the actual security level.
 - Execute emergency exercises and simulate attacks.
 7. Additional technical and organizational actions
 - Monitoring/management of mobile devices, two-factor authentication, segmentation of networks.

The steps are rather basic, but according to CFCS and Agency of Digitisation they are not present in many organizations and as such worthwhile to follow for most. From other guidance released by CFCS, it is noted how cyber defense strategies often are merely a technical solution [9]; this guidance evidently addresses this.

Overall, the guidance [20] is great, easy to read, straight-forward and highly recommended. The most important steps and their content are outlined below:

Step 1 speaks about the importance of top-management support, working with the DS/ISO 27001 standard and questions to answer, like “*Are we convinced that our information are adequately protected?*” and “*Do we have a formal information security policy which we actively support and which our employees understand and follows?*”. These are especially important, as they might uncover errors in the current policies or procedures related to this thesis’ subject.

Step 3 contains the “Top 4” basic security measures, which should be carried out before any other technical measures. The measures with their associated properties can be seen in Table 2.4. It is emphasized that these 3 basic security measures require planning, information across the entire organization and are a necessity before advanced to the next steps.

Step 4 emphasizes the importance of having the technical measures complemented by well-informed employees, because “[*w*]hen the attacks succeed, it is rather due to human error than errors in the systems” [9].

Specifically they need to have knowledge of the attack methods typically used in combination

with “a technical attack” (as [20] puts it). The guidance exemplifies how a social engineer might acquire e.g. information through physical contact, telephone conversations and e-mails in order to gain access to the organization’s assets; in turn, the attacker can utilize legitimate user accounts/-rights to access the organization’s systems – a type of attack, that is “[...] almost impossible to prevent and even detect.” [20]. The guidance states that employees should be made aware of these and similar risks already by the beginning of the employment.

Step 5 also has a great detail on logging of e.g. network activity and security events on local machines. It mentions how and why some organizations might lack here and suggests how to get started properly by e.g. logging only for high-risk targets. CFCS has released a separate guidance on this: “*Logning – en del af et godt cyberforsvar*”⁴³.

Step 6 details how the security measures should be tested regularly and corrected if necessary; examples hereof are pen-tests and exercises (e.g. power take-outs, attacks, hardware fails or back-up plans). This also supports step 4 of creating awareness around IT security in the organization.

Security measure	Employee resistance	Cost of establishing	Cost of maintaining / running	Prevent or detect?	To prevent attack phase 1?	To prevent attack phase 2?	To prevent attack phase 3?
Make a white-list of allowed applications	Medium	High	Medium	Both	Yes	Yes	Yes
Update programs with latest security updates (critical within 2 days)	Low	High	High	Prevent	Yes	Possibly	No
Update OS with latest security updates (critical within 2 days)	Low	Medium	Medium	Prevent	Yes	Possibly	Possibly
Limit number of user accounts w/ domain- or local admin privileges	Medium	Medium	Low	Prevent	Possibly	Yes	Possibly

⁴³https://fe-ddis.dk/cfcs/publikationer/documents/vejledninger_finalapril.pdf

Table 2.4: The top 4 basic security measures as given in [20].

As with the other guides, the we do again see a lack of concrete advice on handling OSINT-data and secrecy hereof. [20] is clear in that it suggests the easiest and most valuable/“lowest-hanging fruits” in terms of IT security measures; we can thus deduce that the content of awareness campaigns should primarily consist of information on current attack scenarios (as per step 4). This makes sense as creating guidelines for a large, heterogeneous group of employees, can prove difficult due to different reaction patterns and experience biases. This is discussed further in Sec. 2.4.

2.3.2.2 “Spear-phishing – et voksende problem”

CFCS has also published a guidance on spear-phishing specifically [8]. In addition to detailing common attack methods, it presents security measures to counter the threat.

The guide both points to the top 4 basic security measures of [20] (see Sec. 2.3.2.1) and additional measures, as the top 4 are not adequately alone. The additional measures are:

1. Prevent user’s access to links, attached files etc. by preventing the mail from reaching the user.
2. Prevent the user from activating the content.
3. Limit the damage should the user activate the content.
4. Establish and activate incident response should an incident occur.

Of interest under these four steps is that the security goal in **step 1** is suggested to be controlled through guidelines containing *expectations* of the user’s conduct. This is in accordance with other guidance from CFCS, where, by describing attack scenarios, we implicitly describe how users should act in response. It is also noted that establishment of (technical) security measures should be complemented by internal education and awareness campaigns.

Step 2 notes the limited technical measures and how awareness again plays a huge role. It should be established through general information campaigns and practical exercises, where real-life examples of e-mails are seen; examples are given in the end of the guidance and detailed in Sec. 2.5.

Step 3 is almost identical to the measures in [20]. It also highly suggests to try to establish a culture of reporting dangerous content received or even activated. This can be difficult, because the user may be afraid to demonstrate his/her ineptitude by having activated malware and maybe get punished for it or submitting a false report. It is adamant to establish a culture accepting errors (of both handling things wrong as well as false reports).

Finally, **step 4** is five simple phases of incident response: Get the right persons, stop the attack, establish emergency operations, clean and reestablish, and evaluation. The last phase here is equivalent to what we have previously seen about regular exercises, also to contribute to awareness in the organization and for revising technical security measures and policies.

2.3.3 NIST (US)

In NIST Special Publication 800-53 “Security and Privacy Controls for Federal Information Systems and Organizations” [55] controls are provided for the same area as found in the controls of DS/ISO 27002 [16]. The content is just as extensive as there, so due to time constraints, we provide a summary of controls deemed relevant to help address issues around outbound information sharing.

The controls are found in Appendix F of [55]. We suggest an effort to harden security with respect to hinder OSINT-data should include the controls as listed in Table 2.5. The controls are widely interconnected and we present the most relevant references for each control to others based on the standard’s list of related controls. Within each standard, a list of enhancements are offered as well; these offer even more advice and related controls and are not included here (except for AC-4).

ID and name	Control	Related controls
AC-3 Access enforcement	The information system enforces approved authorizations for logical access to information and system resources in accordance with applicable access control policies.	(too many to list)
AC-4 Information flow enforcement ad. 5	The information system enforces organization-defined limitations on embedding data types within other data types.	
AC-4 Information flow enforcement ad. 6	The information system enforces information flow control based on organization-defined metadata.	
AC-4 Information flow enforcement ad. 9	The information system enforces the use of human reviews for organization-defined information flows under the organization-defined conditions.	
AC-4 Information flow enforcement ad. 15	The information system, when transferring information between different security domains, examines the information for the presence of organization-defined unsanctioned information and prohibits the transfer of such information in accordance with the organization-defined security policy	

AC-20 Use of external information systems	The organization establishes terms and conditions, consistent with any trust relationships established with other organizations owning, operating, and/or maintaining external information systems, allowing authorized individuals to: a. Access the information system from external information systems; and b. Process, store, or transmit organization-controlled information using external information systems.	AC-3, PL-4
AC-22 Publicly accessible content	Designates individuals authorized to post information onto a publicly accessible information system; b. Trains authorized individuals to ensure that publicly accessible information does not contain nonpublic information; c. Reviews the proposed content of information prior to posting onto the publicly accessible information system to ensure that nonpublic information is not included; and d. Reviews the content on the publicly accessible information system for nonpublic information organization-defined frequency and removes such information, if discovered.	AC-4, AT-2, AT-3, AU-13
AT-2 Security awareness training	The organization provides basic security awareness training to information system users (including managers, senior executives, and contractors): a. As part of initial training for new users; b. When required by information system changes; and c. with an organization-defined frequency thereafter (refer to the control for enhancements with practical exercises and recognizing insider threats).	AT-3
AT-3 Role-based security training	The organization provides role-based security training to personnel with assigned security roles and responsibilities: a. Before authorizing access to the information system or performing assigned duties; b. When required by information system changes; and c. with an organization-defined frequency thereafter.	AT-2
AU-13 Monitoring for information disclosure	The organization monitors organization-defined open source information and/or information sites with an organization-defined frequency for evidence of unauthorized disclosure of organizational information.	PE-3, SC-7

<p>PE-3 Physical access control</p>	<p>The organization: a. Enforces physical access authorizations at defined entry/exit points to the facility where the information system resides; b. Maintains physical access audit logs for-defined entry/exit points; c. Provides organization-defined security safeguards to control access to areas within the facility officially designated as publicly accessible; d. Escorts visitors and monitors visitor activity organization-defined circumstances requiring visitor escorts and monitoring; e. Secures keys, combinations, and other physical access devices; f. Inventories organization-defined physical access devices by some defined frequency; and Changes combinations and keys by some defined frequency and/or when keys are lost, combinations are compromised, or individuals are transferred or terminated.</p>	<p>MP-2, MP-4, PE-4</p>
<p>PE-4 Access control for transmission medium</p>	<p>The organization controls physical access to organization-defined information system distribution and transmission lines within organizational facilities using organization-defined security safeguards.</p>	
<p>PL-4 Rules of behavior</p>	<p>The organization: a. Establishes and makes readily available to individuals requiring access to the information system, the rules that describe their responsibilities and expected behavior with regard to information and information system usage; b. Receives a signed acknowledgment from such individuals, indicating that they have read, understand, and agree to abide by the rules of behavior, before authorizing access to information and the information system; c. Reviews and updates the rules of behavior by some defined frequency; and d. Requires individuals who have signed a previous version of the rules of behavior to read and re-sign when the rules of behavior are revised/updated.</p>	<p>(too many to list)</p>
<p>SC-7 Boundary protection</p>	<p>The information system: a. Monitors and controls communications at the external boundary of the system and at key internal boundaries within the system; b. Implements subnetworks for publicly accessible system components that are either physically or logically separated from internal organizational networks; and c. Connects to external networks or information systems only through managed interfaces consisting of boundary protection devices arranged in accordance with an organizational security architecture.</p>	<p>AC-4</p>

Table 2.5: A minimal list of suggested controls of NIST 800-53 [55] to control outbound information sharing in the organization.

2.3.3.1 Insider threat-study

Also issued by the US government, are the reports on insider threats in critical infrastructure sectors (energy, communications etc.) together with Carnegie Mellon Institute [29, 47]. Mostly containing statistics on a relatively weak population of incidents, an important take-away is to think about access management within the organization itself⁴⁴ to hinder harm – intentional or not – by employees and to establish procedures for correctly issuing access (in [29] an employee were granted unnecessary access to backup tapes) and reporting problematic behavior (if e.g. disgruntled employees express intent to harm).

2.3.4 Centre for the Protection of National Infrastructure (UK)

CPNI is the national centre for physical, virtual and personnel security in the UK. Since the inauguration of the *National Cyber Security Centre* (NCSC) in February 2017, the IT security capabilities previously held here, are however moved to the NCSC. Holistic guidance on all three elements are still found within the CPNI-domain as well as earlier guides⁴⁵.

The holistic guidance provided by CPNI is valuable, as e.g. awareness campaigns and the need to understand social engineering-methods are often mentioned in other guides, but specifics fall outside the traditional, technical IT security domain.

In [11] disruption of hostile reconnaissance is examined. Hostile reconnaissance here defined as *“purposeful observation with the intention of collecting information to inform the planning of a hostile act against a specific target.”* – the more complex attack, the more sophisticated planning and reconnaissance necessary. Even though the organization may face a large variety of threats/different attack scenarios, there will be common features of information requirements between them. Hence disruption of reconnaissance can be valuable countering a wide range of threats and as a security measure overall.

The security manager need to understand the threat in order to counter it. An overview of how to do is given in the first part of the guide, which is examined in Section 2.4.1.

The guidance notes how an attacker may be focused on acting covert and successfully, so the strategy of the defender should focus on giving the attacker the impression of the opposite, by *denying* them opportunity to gain the information, *detect* the reconnaissance and *deter* *“them by promoting failure through messaging and physical demonstration of the effective security”*.

These three principles are the basis for this guidance⁴⁶:

Deny essential, reliable information by ensuring that it is not readily available (i.e. the information that the threat analysis shows are valuable to an attacker). The information should be unattainable online (e.g. removing/modifying information on public websites),

⁴⁴I.e. only issuing administrator privileges as strictly necessary, logging, segregation of networks etc. – in general some of the technical best-practices that may already be employed, but focusing on outsiders.

⁴⁵From <https://www.cpni.gov.uk/cyber-security>

⁴⁶From [11] in a digested version

physically and via people (through awareness); security measures should be non-evident and/or unpredictable (e.g. timing of security patrols).

Detect suspicious activity through integrated, effective capabilities focused on the right areas (e.g. well-placed CCTV or probes in the right spots in the network). Should be unpredictable if possible.

Deter is vital; it is the promotion of the above security measures to change the attacker's perception and assessment of the target and their chances of success. It is a way to maximize the gain from the security measures taken; knowing that not only the security personnel, but virtually everyone is on the look-out, can make the "casual" attacker (non-APT) choose another target instead.

The promotion should be done without revealing important information, but still be credible; a way is to post pictures online advertising new equipment (in general terms! Not mentioning system-critical information) or security measures as a credible, but subtle threat. It is important to be truthful, as the attacker might otherwise discover so, resulting in the entire deterrence strategy losing value.

In [10] CPNI finds that attackers can be discouraged for four reasons:

- A lack of information meant they could not confirm or deny assumptions.
- They could not ascertain detail on organisational structures or personalities.
- A lack of imagery prevented a virtual recon of the physical location.
- The cookies policy included logging of a user's IP address, pages visited and keywords searched for.

Whereas the attacker will be encouraged if:

- Detailed information revealed exploitable weaknesses in security.
- Security did not appear to be a priority for the organisation.
- There was a lack of evidence of physical security measures.
- The website had a bland cookie policy.

From CPNI claims, these eight reasons are thus especially important to follow and can be followed directly by both IT security responsables and advisers.

Having understood the threats and the above principles, CPNI suggests six themes the security manager should to go consider the organization's security for. In addition with them, a checklist is given to answer for each. The six themes are:

- Secure online presence

- Robust entry process
- Hostile reconnaissance threat is understood
- Strong staff security awareness
- Vigilant and professional security
- Deterrence strategy

The checklist is a great resource; it one of the few actual yes/no-lists we have found regarding this thesis' subject. It can be used as an Appendix for a pen-test or scanning report; some of the questions can be viewed as controls or measurements of the controls of the organization. When using the ISMS of DS/ISO 27001, it can be used in addition to the measurements proposed in [17] (see Sec. 2.3.1.2).

It can be seen in its full length in Table C.1 (Appendix C).

2.3.5 National Cyber Security Centre (part of GCHQ, UK)

Originating from CPNI in February 2017 was the NCSC, organizationally located similar to the Danish CFCS under the UK equivalent to DDIS, *Government Communications Headquarters* (GCHQ). “*The NCSC is the single point of contact for the private and public sectors. It brings together the capabilities developed by CPNI and CESG (the information security arm of GCHQ), CERT-UK and the Centre for Cyber Assessment.*”⁴⁷.

NCSC primary guidance on IT security in general⁴⁸ is the “*10 Steps to Cyber Security*” [39] (originally by CPNI from 2012) taking the form of an executive summary; it is complemented by the paper “*Common cyber attacks: Reducing the impact*”[37]. We examine the paper “common cyber attacks” first, at it is more comprehensive.

2.3.5.1 “Common cyber threats: Reducing the impact”

The paper is split into five parts: *The threat landscape, understanding vulnerabilities, patterns of common cyber attacks, reduction of exposure to attacks and case studies*. Only *reduction of exposure* is relevant for this subsection.

It operates with four stages of attack, which is used to present mitigation steps⁴⁹ (all explanatory citations from [37]):

Survey “*Investigating and analysing available information about the target in order to identify potential vulnerabilities.*”; similar to intelligence gathering.

Delivery “*Getting to the point in a system where a vulnerability can be exploited.*”

Breach “*Exploiting the vulnerability/vulnerabilities to gain some form of unauthorised access.*”

⁴⁷From <https://www.cpni.gov.uk/cyber-security>

⁴⁸Their other guides are for very specific subjects like WannaCry, Whaling, Macros in Microsoft Office, Windows XP etc. and presented in a huge mess.

⁴⁹As mentioned in Sec. 2.4 the number of phases and their naming varies greatly.

Affect *“Carrying out activities within a system that achieve the attacker’s goal.”*

In general, the guidance lists a number of “controls” to be implemented. They are not as specific as those found in standards and I suspect the understanding of the term is a bit different. They are similar to the “top 4 basic security measures” [20] and as they are shared among many countries, it could be the UK adoption of them. The controls and their related mitigative effect on each attack stage is shown in Table 2.6.

Further guidance is given for a few of the controls; see [37]. The last three referenced from [39]. No rating is given between which controls to prioritize like [20] does, but this is to some extent indicated by the associated attack stages (a risk assessment could have pointed to which stage should be sought mitigated first).

Security control	Attack stage(s) to mitigate
Establish a network perimeter defense (firewalls, web proxies, web filtering, content checks)	Delivery, breach
Malware protection	Delivery, breach
Patch management	Breach
Whitelisting and execution control	<i>Not mentioned, but most likely:</i> Delivery, breach
Secure configuration	Survey, delivery, breach
Password policy implemented and followed	Delivery
User access control	Breach
Security monitoring	Breach, affect
User training and awareness	Survey, breach
Security incident management	After incident has been discovered/acknowledged

Table 2.6: The “controls” given in [37] on reducing the impact of cyber attacks.

The survey attack phase is the most relevant to this thesis. Of special interest, we find the advice: *“Any information which is published for open consumption should be systematically filtered before it is released to ensure that anything of value to an attacker (such as software and configuration details, the names/roles/titles of individuals and any hidden data) is removed.”* [37] (“hidden data” is meta-data from e.g. documents). This is sound advice, as the aforementioned tool *metagoofil* exactly harvests this kind of information. It can be used to e.g. establish a connection between some employee and an area of work within the organization.

For the survey phase, it is further emphasized how awareness can contribute a great deal; how employees need to be aware of the *“risks of discussing work-related topics on social media”* and each ones potential as a target for phishing. This includes revelation of sensitive information in conversations or to unsolicited phone calls or emails. The advice refers back to [11], which was reviewed in Section 2.3.4.

2.3.5.2 “10 steps to cyber security”

[39] presents additional advice on 10 individual areas of concern in a shorter, executive summary. The areas are almost identical to the “controls” of [37] as listed in Table 2.6.

To this end, the areas *user education and awareness*, *malware prevention* and *home and mobile working* and their advice is interesting to review as well. It is an executive summary, so it is held in general terms; only relevant information not found in [37] (the previous section) is included here, but the 10 steps can provide additional insight to the interested reader.

User education and awareness It is noted how a staff induction process can be valuable. One could imagine the formal and/or verbal presentation of the policies with room for discussion, can enable to information to be better obtained by new hires. Regular “refreshers” of the organization’s cyber security policy can also be valuable. The effectiveness of these actions should be measured and corrected as appropriate.

The need of an incident reporting culture and formal processes are also important, the summary says. The security staff can e.g. emphasize how their work is supported by this and how the organization is helped. This should be done such that the staff does not fear of negative consequences for speaking up against bad practices; formal policies for disciplinary actions support this by indirectly showing when disciplinary action are *not* taken.

Malware prevention Apart from technical advice identical to that of [37] on measures like end-point control, filters etc., this step includes specific awareness steps for employees to take:

- Think before clicking, and report as soon as possible if you did.
- No use of unapproved removable media/devices.
- Report strange/unexpected behavior (both virtual and physical, it is understood).
- Keep updated on the incident reporting process.

These points are good reminders to include in an awareness programme and as “controls” (in the term of NCSC) in the company.

Home and mobile working This step further expands advice on user awareness. It is important that the user is also aware, that they are expected to look after their mobile devices at all times, be aware of eavesdropping/onlookers, store credentials and devices securely and report any incident.

Technical advice within this area of concern is of course also given, but not relevant here.

2.3.6 Federal CIO Council (US)

The working group Web 2.0 Security Working Group under the authority of the Information Security and Identity Management Committee (ISIMC) as chartered by the Federal CIO Council

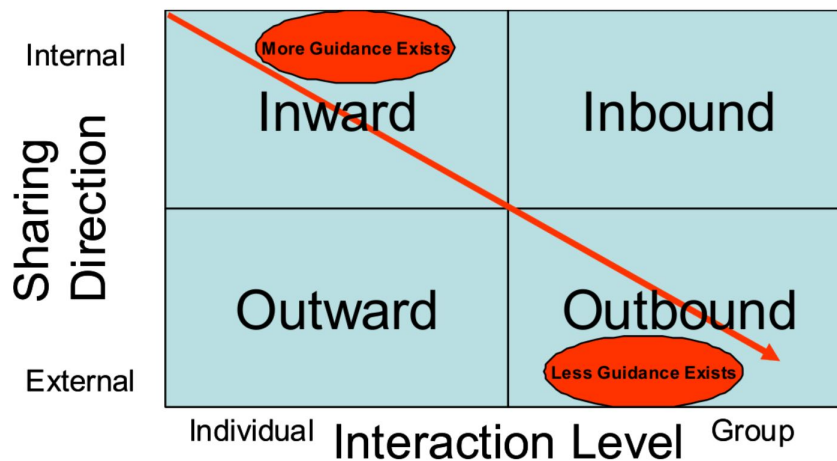


Figure 2.9: The four use cases of social media use of [25] and the amount of associated guidance for them. From [25].

(FCIO) of the US, has published “*Guidelines for Secure Use of Social Media by Federal Departments and Agencies*” [25]. These guidelines are created with the intent to minimize the risk coupled with initiatives under President Obama to communicate with and include the general population using e.g. social media.

The guidance gives “*recommendations for the creation of a government-wide policy for social media, addressing policy controls, acquisition controls, training controls, and host and network controls*” and do so with a general, non-vendor or -technology specific approach.

The guidance identifies four use cases for social media within the guide’s context of federal organs. They are depicted in Figure 2.9 and described as (from [25]):

Inward sharing is “[...] *sharing of internal organizational documents [data] through internal collaboration sites [...]*”.

Examples: Internally hosted SharePoint or wiki’s.

Inbound sharing is exemplified with crowdsourcing.

Example: Change.gov (used for directing questions and proposals directly to the administration of the US).

Outward sharing is federal information to be shared with e.g. “*state and local governments, law enforcement, large corporations, and individuals.*”. Also called “inter-institutional sharing”.

Examples: Agency communication to the public using social media during emergencies or STAR-TIDES, a knowledge sharing research project.

Outbound sharing “*is federal engagement on public commercial social media websites.*”.

Example: Interaction by a Secretary of State with foreign media through Twitter.

The use cases are referenced from Gartner Research, but this source cannot be identified currently. The use cases generalizes what type of social media interaction exist. They can be useful for

distinguishing between advice for different scenarios such that guidance subsequently can be presented in the right scenarios and to the right persons.

The guidance notes how inward sharing has a lot of guidance already, including standards with associated controls, whereas “[l]ess federal guidance exists for inbound, outward, and outbound sharing use cases, and the guidance that does exist is relatively recent.” [25].

The recommendations of the guidance is split into controls for policies, acquisition, training, network and hosts; relevant parts are included in the following subsections. They interested reader should review them in full in the original guide.

2.3.6.1 Policy controls

This Section notes that the safe use and navigation of social media as behavioral issue, not a technology issue. This is because users will find a wide variety of platforms and these are subject to constant changes. Thus policies should be in place regulating access and distribution of data in both personal and private settings.

Federal agencies following this guidance are to develop guidelines for social media; one such example from the US Air Force is found in [12]. Directions to specific NIST publications concerning risk assessment and other parts also seen in the ISMS of DS/ISO 27001 is given; these are included in Sec. 2.3.3.

2.3.6.2 Acquisition controls

Apart from technical controls and controls directed at specific services, this Section suggests a special rule-set for using e.g. .mil or .gov-addresses. They primarily aim to have the social media hosting provider implement this, but evidently a policy on using such highly specific addresses would also be valuable to adopt. Such an address is supposedly the employee’s organizational email address and thus necessary to use in most cases, but there is no need to use it to for the contact sheet of the local brass band. If it has to be used on social media, the control says how details of employment/work, location, resume, skills and similar should not be included (I believe this should be extended to most info really, because private information can also quickly become a valuable stepping stone for a social engineer to gain confidence through e.g. “shared interests”).

2.3.6.3 Training controls

This guidance is very critical of users ability to protect sensitive information; it reads:

“Users are almost always the weakest link in an information system, and may inadvertently divulge sensitive information through a social network. Few effective technical security controls exist that can defend against clever social engineering attacks. Often the best solution is to provide periodic awareness and training of policy, guidance, and best practices. The proper use of social media [...] should be part of annual security awareness training.” [25].

Specifically, the training controls should include:

- An official policy/guidance on use of social media. The US Air Force’s “New media guide” [12] is given as an example; to the reader, it can be used as inspiration, but no specific advice usable to this project is found in it.
- Training employees “[...] to be mindful of blurring their personal and professional life” and to not engage with external professionals that might do the same.
- Guidance on how to present themselves online; similar to the previous Section on acquisition controls, some roles may require disclosing some details, while others are not in a position to do so.

In addition, this Section suggests working with the organizational culture and general awareness training of cyber attack scenarios as also e.g. CPNI advocates (see Sec. 2.3.4).

2.3.6.4 Network and host controls

This Section only contains technical controls. It suggests use of both US-specific federal solutions and more regularly available technologies like establishing a Security/Network Operation Center (SOC/NOC), web filtering, network segregation, use of strong authentication, sandboxing (executing files on a virtual machine to avoid infections) etc. It is not relevant here.

2.3.7 Agency for Digitisation (DK)

“The ‘Agency for Digitisation’ [Danish: Digitaliseringsstyrelsen] is an agency under the Ministry of Finance established in 2011 to speed up the digitisation processes required to modernise the Danish welfare society. The Agency is in charge of the digitisation of Denmark and is responsible for the implementation of the government’s digital ambitions in the public sector.”⁵⁰

Specifically, they provide guidance on many aspects of the digitisation including information security. From their knowledge base on *external requirements for information security* [19], they list a number of laws applicable to public government institutions, but also other organizations can benefit from the agency’s collection.

As has been seen in other sources, we find that the laws are relating to “direct” assets, and not OSINT-data of e.g. the systems or the employees themselves. If we wanted to be able to relate to the specific causes of an outbound flow of data to OSINT-sources, we have to interpret the laws and do so in the environment of specific organizations. The proficiency from real life positions and expertise this requires is not held by the author and thus out of scope of this thesis.

The relevant Danish and EU laws applicable to the data as subject of this thesis from [19] are:

- *Lov om behandling af personoplysninger* (“Persondataloven”) — Law no. 429 of May 31st 2000 latest changed by law no. 519 of June 6th 2007 (Department of Justice/the Danish Data Protection Agency) <https://www.retsinformation.dk/Forms/R0710.aspx?id=828>

⁵⁰From <https://www.digst.dk/ServiceMenu/English/About-the-Danish-Agency-for-Digitisation>

- *Bekendtgørelse om sikkerhedsforanstaltninger til beskyttelse af personoplysninger, som behandles for den offentlige forvaltning* (“Sikkerhedsbekendtgørelsen”) — Decree no. 528 of June 15th 2000 latest changed by Decree no. 201 of March 22nd 2001 (Department of Justice/the Danish Data Protection Agency) <https://www.retsinformation.dk/Forms/R0710.aspx?id=842>
- *Lovbekendtgørelse af forvaltningsloven* — Decree of law no. 433 of April 22nd 2014 (Department of Justice) <https://www.retsinformation.dk/Forms/r0710.aspx?id=161411>
- *Information security laws for NATO-countries — NATO: “Security Within the North Atlantic Treaty Organisation”*, CM(2002)49 1th, <http://www.statewatch.org/news/2006/sep/nato-sec-classifications.pdf>

Although the two first laws primarily contains regulations for handling sensitive personal data, they have a general implication in connection with the design of policies, procedures and safety measures and thus the design of IT-systems and their use and concrete risk assessments of these. The two latter are concerned with professional secrecy of government organizations for different cases, but are only relevant in select cases of either organizations handling customer data respectively organizations handling NATO-information.

2.3.8 Other sources

The sources in this subsection are not relevant to the project following the introduction to this part, but can either be a great source of information to the organization or, for laws/decrees, be applicable to a few select organizations:

- As referenced in Section 2.3.6, the US Air Force has a social media policy from 2009 [12], which can serve as inspiration to an organization to develop their own.
- In his book “Art of deception” [35], former black hat-hacker and social engineer Kevin Mitnick presents a number of controls based on his own experience with first exploiting these vulnerabilities and later work with organizations to eliminate them. He provides best practices based on his own experience. Among others, these include a flowchart for determining eligibility of an information or action request, information security policies/procedures for organizations (e.g. how to verify and authorize a person) and suggestions for an awareness programme. It also includes a list of factors, which can contribute to make organizations more vulnerable to attacks, which can be referenced for e.g. content of awareness programmes (large number of employees, multiple locations, lack of awareness, no formal procedures etc.).

Whilst not being a standard to follow, the policies are all exemplified and with social engineering in mind such that their purpose is evident. They are however created before the DS/ISO 27000-series, so they can deviate from current best practices.

Of particular interest in a social engineering-perspective are the following policies (all after [35], but shortened):

- 5-1 IT department employee contact information** Phone numbers/e-mail addresses of individual (IT department) employees should only be disclosed on a need-to-know basis to prevent abuse by social engineers.
- 7-8 Generic e-mail addresses** Use generic e-mail addresses for departments ordinarily communicating with the public. This prevents social engineers from associating employees with specific tasks/departments for easier profiling.
- 7-11 Contact information on Web sites** No details of organization structure or employee's names/contact information shall be shown on the organization's web site. As the above, this enables the social engineer to sound knowledgeable about the organization or plan the attack better.
- 9-8 Personal identifiers** All personal identifiable information of employees should never be used as identifiers. These can be harvested/bought purposefully by the social engineer (as such they should not be found on from OSINT-research either).
- 10-10 Posting company information online** No details of the organization, its inner workings, hardware/software, contact information etc. shall be disclosed online on public forums and similar other than in accordance with company policy. Forums and similar are very useful resources to a social engineer whom might e.g. find a post on a current issue or, from an account know to be associated with an employee, learn details of the organization's inner workings and tools.

The above are only some specific examples; the book contains many other great examples of specific policies to implement (or at least consider) for management (Section 1), IT (Section 5, 6, 7, 8, 20), HR (Section 17), employees in general (Section 9, 10, 11, 12, 13, 14, 15, 16) and guards/security personnel/reception (Section 2, 3, 4, 18, 19). Not all of the above fits with the structure of the auto-generated report; it is e.g. not possible with Maltego to determine if policy 9-8 is violated.

- In [27] Interpol presents a list of recommendations for individuals respectively organizations to take to hinder social engineer fraud.
Individuals are advised to *“Remain vigilant and take the time to assess any e-mails you hadn't expected to receive.”* and take common steps to check the e-mail's legitimacy. Phone calls are also warned against.
For organizations, it is advised to implement proper procedures in the organization, promote awareness and establish formal points-of-contact with authorities and financial institutions (and make sure that the financial institutions are also vigilant by e.g. requiring two-factor authentication for transfers).
Finally, steps to take after compromise is given for both private entities and organizations; it is advised to contact financial institutions, change compromised passwords and to document and report all interactions.
- It would have been relevant to look at *The General Data Protection Regulation* (GDPR; EU Regulation 2016/679) going into effect May 25th 2018 (in Denmark), but the Danish

interpretation by the Department of Justice⁵¹, was only publicized on May 24th 2017⁵², which is way too short time to read 1090 pages, which even large organizations will need some time to chew through and implement⁵³. Concrete guidance are not planned to be released until September 2017 and through to January 2018⁵⁴.

It is not expected to fundamentally alter current Danish law⁵⁵ as listed in Section 2.3.7, but will require most organizations to have a “Data Protection Officer” (DPO) to act as a single point of contact regarding data protection and the law, whom should be able to demonstrate compliance. With the regulation also follows fines of up to € 20 million or 4 % of annual turnover, whatever is smaller.

2.4 Social engineering-techniques

For an organization to act proactively to the threats facing them, it is necessary to understand what kinds of attackers/threat scenarios they face and how the attackers act.

This section looks at the general attack phases and techniques an attacker can employ in attack scenarios relying (in part or fully) to exploit human psychology and employ OSINT-data in this; an attacker using these techniques can be referred to as a *social engineer*.

The next section (Sec. 2.5) will, based on the general description found here, list a common cyber attacks with focus on the OSINT that enabled them. The information can then be used as a reference for the auto-generated report, so that the organization can see how some piece of data found in a pen-test relates to an actual threat towards them and specifically how the scenarios may elapse.

2.4.1 The attack-phases

The general phases of attacks with an element of social engineering are sought to be presented in here. This section can be used as specific guidance on the threat modeling-phase of a risk assessment or pen-test (see Sec. 2.1.3).

An attack is typically modeled as a number of phases (3-7 depending on the source like certification bodies or standardization organs): Reconnaissance/survey, weaponization/customization, delivery, penetration/breach/break-in, enumeration/installation/maintaining access, data export and “covering the tracks”. Some of the papers already examined here describing attack phases are

⁵¹All parts will be available here: <http://justitsministeriet.dk/nyt-og-presse/pressemeddelelser/2017/nye-regler-styrker-beskyttelsen-af-persondata-i-europa>

⁵²See <https://www.kromannreumert.com/Nyheder/2017/05/Betaenkning-om-persondataforordningen>

⁵³A couple of citations from companies, that think it is way too short time to react is found in this article: <https://www.computerworld.dk/art/240087/bliv-klar-til-eu-persondataforordningen-ministerium-klar-med-laenge-ventet-dansk-vejledning-til-den-nye-eu-lov>. See also e.g. <https://www.danskerhverv.dk/Nyheder/Sider/Betaenkning-om-persondatabeskyttelse.aspx>

⁵⁴http://justitsministeriet.dk/sites/default/files/media/Pressemeddelelser/pdf/2017/plan_for_vejledning_om_forordningen.pdf

⁵⁵According to e.g. <https://www.bdo.dk/da-dk/faglig-info/advisory-publikationer/forensic-assurance/hvordan-paavirker-eu-persondataforordningen-dansk>

[20, 7, 37, 24].

One specific model used in some official US publications is the “Lockheed Martin kill chain” (depicted in Figure 2.10). The figure both depicts the phases of an attack (noticeable similar to CFCS’s used in e.g. [7]) and the phases of disruption (quite similar to the one used by CPNI in [11]). Sources describing fewer phases can still encompass into this model using synonyms as e.g. the ones above. The attacker will use various tools to reconnoiter his target before the actual

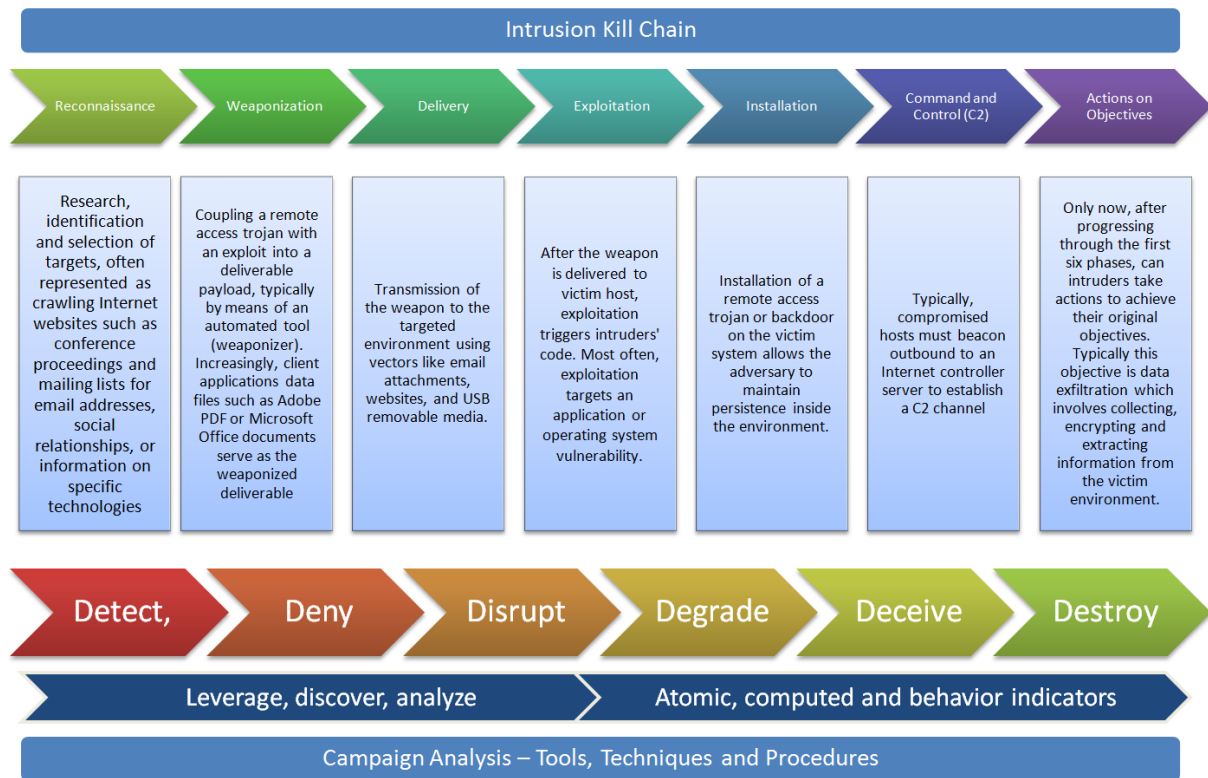


Figure 2.10: The “Lockheed Martin kill chain”, from <https://countuponsecurity.com/tag/kill-chain/> after [24].

attack; social engineering plays a large role in this phase, but the techniques can definitely also be utilized during the exploitation phase to gain further information, access etc.; techniques are described in Section 2.4.2.

The goal of the reconnaissance phase is “[...] to obtain enough detailed information and get sufficient certainty about the reliability of this information to inform their modus operandi [‘habits of working’] and be sure of success.”[11].

Specifically for a social engineer, we find in [35] “The Social Engineering Cycle” which describes the steps the social engineer will repeatedly cycle during an engagement (the same steps are used in [27]):

Research OSINT gathered from government sources, commercials, news sources, web content and even physical acquisition (like “dumpster diving”).

Developing trust Using information from the research, insider information from previous steps, misrepresentation, citing need for help, citing people known to the target or authority.

Exploiting trust Asking for information or an action from the target (or manipulate the target to ask the attacker for help).

Utilize information Repeat the cycle if necessary.

The cycle can be seen as both an alternative to the reconnaissance, attack and exploitation phases of e.g. the kill chain (Figure 2.10) and as a specification of the exploitation phase of the same, as “The Social Engineering Cycle” is the specific method to exploit the target.

Another entrance to understanding the attacker’s reconnoitring methods is to look at his mindset. It can be characterized by *intent*, *capability* and *culture* [11]⁵⁶:

Intent This is what the hostile wants to achieve. Think about their overall aim as this will help identify the effect the hostile wants the particular attack to have.

Capability This is about the resources at the hostile’s disposal. Think about equipment, time, personnel, skills and training, financial backing and geographic location.

Culture This is the hostile’s personal motivations and appetite for risk.

Even though not all dimensions of the attacker can be defined, the organization will have gained further insight and are able to determine likely attack scenarios. With this in hand, the organization can move to the actual threat modeling (see Sec. 2.1.3) for a framework to build the scenarios.

Bear in mind, that while most attackers are outsiders, as many as 30 % of attacks may come from insiders with prior organizational knowledge [29]. Insiders may be required to gather far less information to deploy a successful attack. These should be covered when performing threat modeling, as intent often vary from the outsider; in [29] the motive was most often sabotage, while many current attacks (at least the ones reported in media) seeks a monetary gain or access to intellectual property/espionage. Insiders can however be harder to profile; [29] finds that no “demographic profile” exists of a malicious insider.

Having build them, it is important to revisit and update them often to their maintain usefulness [11]. Attackers (especially highly motivated ones, e.g. APT-groups) will find new ways and changes in the organization may have gone unregistered.

In [35] p. 332 a list of “common social engineering methods” are shown; it is reproduced below. Several has already been covered from other sources’ examples of attacks (especially in Sec. 2.3), but these are made by an experienced social engineer. In Section 2.4.2 we will also see how many of these methods relate to specific psychological weaknesses to exploit.

The list can be helpful both in modeling attack scenarios and for awareness programmes.

- Posing as a fellow employee.
- Posing as an employee of a vendor, partner company, or law enforcement.

⁵⁶Descriptions of the three characterizations are also from [11].

- Posing as someone in authority.
- Posing as a new employee requesting help.
- Posing as a vendor or systems manufacturer calling to offer a system patch/update.
- Offering help if a problem occurs, then making the problem occur, thereby manipulating the victim to call them for help.
- Sending free software or a patch for victim to install.
- Sending malware as an e-mail attachment.
- Using a false pop-up window asking the user to log in again or sign on with password.
- Capturing victim keystrokes with expendable computer system or program.
- Leaving a CD/USB around the workplace with malicious software on it.
- Using insider lingo and terminology to gain trust.
- Offering a prize for registering at a web site with username and password.
- Dropping a document or a file at company mail room for intraoffice delivery.
- Modifying fax machine heading to appear to come from an internal location.
- Asking receptionist to receive the forward a fax.
- Asking for a file to be transferred to an apparently internal location.
- Getting a voice mailbox set up so callbacks perceive attacker as internal.
- Pretending to be from remote office and asking for e-mail access locally.

We call these actions *pretexting*⁵⁷, as the social engineer proposes some of the above reasons to perform his malicious requests or actions. [35] contains several more examples of stories of attacks performed by both Mitnick himself and “not-Mitnick”; he has also published several books on this (and so has a lot of other people).

2.4.2 The psychological tricks of a social engineer

In this section, we describe specific “tricks” employed by a social engineer in common attack scenarios. It is a bit out of scope of this thesis (and the study line), so it is kept on a general level just to get a basic idea of the techniques. Concrete examples of their use in cyber attack scenarios are presented in Section 2.5.

⁵⁷See <https://en.wikipedia.org/wiki/Pretext> for a definition.

The definition of *social engineering* is in [37] given as a technique for “[...] *fooling people in to breaking normal security procedures.*” and carried out by anyone. In [27] it is stated that the term “[...] *refers to the scams used by criminals to trick, deceive and manipulate their victims into giving out confidential information and funds.*”

In “The Social Engineering Cycle” above we see how the social engineer can acquire information using tricks to develop and exploit the trust of a victim. The social engineer needs to be aware of how he can affect the human into willingly perform whatever act he wants him to.

The human mind perceives and processes a lot of information unconsciously all the time. Understanding these techniques enables a social engineer to exploit his target unknowingly or heedlessly. Much research has gone into this. Of the more dubious kind are the research on *neurolinguistic programming*⁵⁸ (NLP), which claims it is possible to “re-program” the brain to achieve some goal (both personal to imitate successful people, treat problems and manipulate others; the last is called *dark NLP*). The Wikipedia-page has numerous references to studies and sources discrediting this.

Daniel Kahneman is a renowned psychologist and professor knowledgeable especially in decision making and behavioral economics, for which he in 2002 received the Nobel Prize. In [28] has collected his work (performed together with late Amos Tversky); of interest here is the work on decision making and cognitive biases, which an social engineer can exploit to trick his target’s perception of some event.

Kahneman describes the brain’s thinking in two systems⁵⁹: System 1 acts “*Fast, automatic, frequent, emotional, stereotypic, subconscious.*”⁶⁰ “[...] *with little or no effort and no sense of voluntary control.*” [28], while System 2 is “*Slow, effortful, infrequent, logical, calculating, conscious.*”⁶¹ and “*allocates attention to the effortful mental activities that demand it [...]. The operations of System 2 are often associated with the subjective experience of agency, choice, and concentration.*” [28]. Kahneman notes how “[...] *System 2 believes itself to be where the action is [...]*”, but actually System 1 is origin to the “[...] *impressions and feelings that are the main sources of the explicit beliefs and deliberate choices of System 2.*”

The engagement of System 2 requires attention and when drawn away, System 2 is disrupted. A famous example hereof is the experiment “The invisible gorilla” by Simons and Chabris⁶² in which two teams passes balls between each other. The viewer is tasked to count the number of passes between the players of team 1 and ignore team 2; a task requiring attention put specifically on team 1. At some point during the video, a gorilla enters the stage for 9 seconds. Only about 50 % of viewers notices the gorilla.

[28] explains how this is an example of attention being allocated to System 2, but only towards one of the teams. In turn, System 2 is not available to process the basic input of the automatic

⁵⁸https://en.wikipedia.org/wiki/Neuro-linguistic_programming

⁵⁹Further descriptions and examples can be found on both the referenced Wikipedia-page and in [28]

⁶⁰From https://en.wikipedia.org/wiki/Thinking,_Fast_and_Slow#Two_systems

⁶¹From https://en.wikipedia.org/wiki/Thinking,_Fast_and_Slow#Two_systems

⁶²Two different versions of the experiment video are available on their homepage: <http://www.theinvisiblegorilla.com/videos.html>

functions (seeing and orienting) of System 1. In the experiment, viewers would even deny a gorilla to have entered the stage, which illustrates how we can both “[...] be blind to the obvious, and [...] also blind to our blindness.” [28].

In our everyday life, people often try to save energy, both physical and psychological. This also goes in terms of the two Systems, were we rely on System 1 (acting unconsciously and automatically) to give input to System 2 as described and often even process the input to save attention. It is this “function”, a social engineer might exploit in a number of ways.

There are many concepts in [28]; they are grouped into three sections: “Heuristics and biases”, “overconfidence” and “choices”. All the concepts are closely related and there are way too many to describe here. Some of those that could be exploited, are summarized here:

Science of availability Judgment from how easy examples are brought to mind. Recent plane crashes can make people afraid of flying or the fewer examples can be thought of when judging own personality traits, the higher one ranks himself. Awareness campaigns actually targets the same: By presenting employees with a personal example of social engineering, they expect it as a possibility if they experience it in person.

Bad events/loss aversion The brain is wired to perceive a threatening face in a crowd of happy people, but not the other way around. An experiment showed the brain subconsciously went into “alert” having been showed a threatening set of eyes for $\frac{2}{100}$ of a second without System 2 was aware. Emotionally threatening words attract way more attention than “happy” words, i.e. a threat or a sense of urgency will weigh heavily when deciding on a course of action.

We are also much more prone to avoid bad self-definitions than to pursue good ones, because System 1 processes them on behalf of System 2. This is leads to that *“Bad impressions and bad stereotypes are quicker to form and more resistant to disconfirmation than good ones.”* [28].

WYSIATI The concept *“What You See Is All There Is”* (WYSIATI) covers covers a large range of biases originating from the Systems seeking coherence of information, but not completeness. It is a variety of judgment biases, including:

Confirmation bias To search for information that confirms pre-existing hypotheses.

Overconfidence Used to create sense of a complex world, the mind puts to much trust on the information at hand and suppress doubt of vital information missing and ambiguity.

Framing effects E.g. how 90 % survival rate sounds more promising than a 10 % mortality rate when standing next to your kin in a hospital bed.

Base rate neglect An example given is a description of a quiet and tidy male; when asked whether he is most likely to be a farmer or a librarian, people answer librarian due to the description of the person – even though there are 20 times more male farmers than librarians.

Later WYSIATI is also exemplified as “[...] *constructing for the best possible story from the information available.*” [28].

Intuitive predictions A person is presented with some evidence and proposed a “target of prediction”. He will seek to create a link between the two, using concepts of WYSIATI and one’s norm/perception of some subject. Kahneman notes how surprisingly almost everyone has a perception of even the most obscure subjects, e.g. how a professional sports team manager will think during the game or when selecting players. People will not be able to point out how this norm was created, but it is there.

System 1 uses the associative memory here; it can reject false information, but smaller inconsistencies it cannot distinguish – “*as a result, intuitive predictions are almost completely insensitive to the actual predictive quality of the evidence.*” [28].

From the examples above, we can set actual social engineering tricks into context and get an understanding of why they, despite awareness efforts, work.

Intuitive predictions and WYSIATI is a strong driver in this. One can imagine how a social engineer might place a call to a target, claiming to be some specific person and present evidence mostly resembling what the target requires to hear/know to **intuitively** fit the social engineer with his claim. He might have been *overconfident* in connecting the evidence in a complex statement or call for action or by using *confirmation bias*, expecting to be the receiver of the call or having heard a plausible claim of identity, believing to understand the situation and the evidence given (e.g. a story or a fraudulent e-mail); **intuitively** he connects the two.

A list of “common social engineering methods” from [35] p.332 is shown in Section 2.4.1. We can see how some of the methods can work in the context of Kahnman’s theories; e.g. using *insider lingo* will enable the target to create a false picture of the social engineer of being “one of us”. The attacker can also *pretext* his target pretending to *call to offer help or a systems update*.

The reason this method works following the theory of Kahneman, is that the target intuitively will create a concept of reality, because “WYSIATI”: He will not consider the information that is not there (maybe incited by the social engineer using other tricks); the target is not only blind to what might to others appear obvious, but he also blind to the fact that he is blind in the first place – but it is convenient to the target: He as has reduced the question from “Who is he and is he allowed access?” to “Could he be a peer?”. In turn, this also helps strengthens the social engineer’s proposed story, as the target now has inferred the situation and identity of the social engineer himself instead of being told by him. This is creates a stronger case in the target’s mind.

Similarly, the list “warning signs of an attack” also contains methods, which we can relate to the concepts of [28]: Stressing urgency, threat of negative consequences, claim of authority and name dropping all contribute to establishing a context of high pressure/importance, which will lead the target to comply to **avoid bad events**.

Another theory can also explain to the attack method of cheating the target into thinking he is

interacting with a peer: The attacker can aim to create an aura of “*belongingness*”⁶³.

From the Wikipedia-site on the topic, several studies are referenced on how people like to feel related to and understood by a counterpart. The drive to form and maintain social bonds is very strong and hence the feeling of “social relatedness” is associated with positivity and may also enhance feelings of self-worth [60]. On a similar note, *conformity* to a group is also important to an individual, so his actions can be influenced, if it is possible to trick the target into thinking that he is acting outside the group norm.

Methods of the social engineer as listed in Section 2.4.1 contains examples exploiting the above traits. By posing and acting as a fellow employee (using insider lingo) or name dropping, the attacker does exactly seek to create this feeling of “belongingness” with his target and subsequently gain his trust.

Finally a theory named the “*foot-in-the-door technique*”⁶⁴ can be used. It explains how a person is more inclined to agree to a large request, if he is posed with a “lesser” request first (smaller in e.g. work-load, cost or intrusion of privacy). [35] has a couple of examples of this, where e.g. the social engineer asks an employee for a bit of time to answer a survey and afterwards calls back to ask the employee to extract data for him (e.g. to print and mail an internal e-mail or read a guard duty schedule).

2.5 Common OSINT-enabled attack scenarios

The aim of this section is to provide the reader with an overview of common cyber attacks enabled by OSINT-data.

Weight here is put on demonstrating a wide range of common cyber attack scenarios⁶⁵ with emphasis on what types/sources of OSINT-data enabled the attacks (and not as much on how e.g. the exact conversation between attacker and employee unfolds or the methods involved; this is found in Section 2.4 where we (among others) present a list of common methods by social engineer Kevin Mitnick from [35]).

If an attack does not require the attacker to employ OSINT-data, we do not consider it here. OSINT-enabled attacks works in parallel with “technical” attack methods [20], but the technical parts are omitted here. Physical interactions are often also employed, but as they are only briefly described here, as they are both highly dynamic in nature (thus hard to digest on paper) and also relies heavily on psychological tricks, which are out of scope for this thesis; in Section 2.4.2 the interested reader can find a brief overview of such techniques.

The overview enables us to in general what types of information are crucial to their success. In turn

⁶³See https://en.wikipedia.org/wiki/Belongingness#Group_membership (accessed 2017-07-03)

⁶⁴See e.g. <http://io9.gizmodo.com/these-two-psychological-tricks-will-get-people-to-do-yo-513064707> or <https://blog.enhancv.com/8-persuasion-techniques-to-change-anyones-mind/>

⁶⁵Examples of attacks are included in several of the guides examined in Sec. 2.3; additional have been found in sources, which were not in that section, news articles and some popular books on the subject.

we can then point out in what way(s) an organization's data found on OSINT-sources makes them vulnerable to specific cyber attacks, which is the goal of the auto-generated report (see Sec. 1.1.1).

Performing an OSINT-enabled attack means that the attack uses publicly available information⁶⁶) to create and target the attacks.

OSINT is *all* information that can be accessed publicly; no reference is made on authentication or secure access by the sources I have examined, so it is assumed that *publicly* entails that mostly anyone can be granted access to the information.

The information may be generated as *footprints* of the organization and its employee's daily operations (e.g. from public registers (government or 3rd party)), as a product of use of IT systems, as web content (e.g. articles, documents and their meta-data), news or active information sharing by individual employees on e.g. social media and fora.

To find the information, the attacker can use search engines like Google and Shodan, but also the organizations' own sites, government sites or public registries; tools described in Sec. 2.1.2.2 can support this search.

The information found is then utilized to try to exploit human psychological mechanisms to deceive and manipulate their targets (i.e. *social engineering*) to achieve their goal [27], by e.g. establishing context with the victims such that they place an unmerited degree of trust on an object/subject (see Sec. 2.4.2 for further examples of these tricks).

We cannot claim or guarantee a comprehensive list of attack types or OSINT needed for specific scenarios, but from the sources examined and news articles, we get a sufficiently broad and differentiated view to make the auto-generated report valuable.

The lack of a "complete list" of attacks are due to the inherent opportunistic nature of many cyber attacks (especially the automated ones hunting the low-hanging fruits) and also the fact that many attackers will keep their specific methods secret to avoid detection. And even when a ruleset might have been implemented to detect a certain threat, the attacker will maybe only need to alter his methods slightly. In turn, this creates a "new" attack type. Many of the described attack types mix well for even more advanced attacks.

To help ensure we cover a wide range of attack types, we could use some type of threat modeling. Characterizing different aspects of the attacks, may help thinking about different ones. Unfortunately, this requires some assets/a target organization to follow the described methodology of Section 2.1.3 [11].

Of the sources found for the previous sections, only two (NCSC and Interpol) categorizes attack types without a subject organization: [37, 27] distinguishes between targeted and non-targeted attacks. This is a good distinction, because methods differ a lot between automated and manual attacks. It can also explain how the attacker might find the information for the attack. This

⁶⁶A definition can be found in [53], which is referenced by both NIST and NCSC, but it may be a bit strict and not adhere to all people's perception of what OSINT is; Danish CFCS simply puts it as all public accessible data: <https://fe-ddis.dk/Opgaver/Efterretningstjeneste/Pages/Efterretningstjenesten.aspx>.

is opposed to e.g. making a notion of difference between attacker type/capabilities (APT's, "hacktivists" and cyber criminals), which all may target both specific organizations and perform un-targeted attacks depending on agenda of the individual attacker. This is employed by CFCS in their guides [9].

Other sources list examples of attacks instead. Combined with the list and categorization from [37], we can compile an overview of common attack types. Some notions of attack types (like phishing and ransomware) are used in a very broad sense, where some of the sources here may have a more specific definition. We try to keep their definition and take in other examples as necessary.

Bear in mind when reading these examples, that attacks can both come from outsiders and insiders with prior knowledge of the organization [29]. Insiders may be required to gather far less OSINT (if any) to complete a successful attack due to existing knowledge, so in these cases, the given examples of OSINT to employ for the attack, are not accurate.

The threat of insiders are best mitigated by proper operating procedures and controls as presented in Sec. 2.3.

2.5.1 Targeted attacks

A targeted attack is an attack, where the attacker is specifically interested in the target. Reconnaissance can take months and the attack is tailored to the target. Common targets of attacks are those unaware of the value of information (receptionists, administrative assistants, security), those with special privileges (IT administrators/help desk), manufacturers/vendors (of hard-/software used in the organization) or specific departments (HR, accounting).

Information may be found both using commodity tools, sources (OSINT) and methods, but also closed sources or active social engineering-methods (both online and physical) using the previously found information to retract data from people. The attacker is also called an APT or a state-sponsored attacker⁶⁷.

Examples of attack scenarios of this type are:

Spear-phishing This attack is carried out mostly through emails as the easiest attack vector, but also phone calls, face-to-face or through other means of communication [8] (as people may recognize voice or bi-modalities of the impersonated person/organization); also called *pretexting* in some sources [27].

The goal is either information disclosure for further attacks (any goal, e.g. brute forcing logins to common web services) or to directly e.g. get a monetary gain (through encouraging bank transfers, acquire passwords, (bank) account information or NemID-keys – see also the entry on “regular” phishing in Section 2.5.2) or delivery of a attack payload for e.g. espionage or activism or any other goal.

⁶⁷As explained in the introduction, this is groups with virtually unlimited resources of knowledge, manpower, money, time etc.

The most important differences from un-targeted phishing attacks, is that they target a few, specific receivers, put more work into creating a credible email/relation through language, logo's, current activities/contacts of the organization and non-threatening content. However, while they may seek to imitate language of e.g. a professional email or invoice, another trait used in the emails is a sense of urgency and/or secrecy to convince the receiver to perform the task fast (e.g. a bank transfer) and without disclosing anything to colleagues [8, 38].

To improve credibility, the attacker can employ OSINT to discover:

- Current professional relations (e.g. suppliers, collaborators or customers) found on e.g. LinkedIn, Facebook, public forums, job advertisements (describing technical qualifications needed of new hires) or homepage of the organization or their vendors/-customers.
- Private relations or economic interests found on the aforementioned sources or through e.g. public leak data including company domain email addresses [25].
- Employee names and private e-mail addresses (from e.g. social media accounts) to deliver a malicious payload circumventing organizational countermeasures [37].
- Specifics of the organization's structure from e.g. informative organizational chart, job postings, points of contact (for homepage, support or legals) or meta-data from documents on the organization's homepage. Specifics can include names, positions, job titles, phone numbers, location⁶⁸ etc. Phone numbers can also serve as an alternative contact medium, where the attacker will then employ other parts of the collected OSINT.
- Knowledge of organizational operations (in addition to the previously mentioned) like travel plans, current issues (from public forums or bug reports) [25].

The attacker can of course also employ technical solutions to increase credibility by e.g. acquiring access to email servers [7]. This requires prior use of social engineering to gather passwords, deliver malicious payloads or similar.

Examples of attack are invoice fraud⁶⁹ with fake invoices looking to come from real vendors, coaxing employees into sending money to "colleagues"⁷⁰, delivering malicious payloads [7] or trying to get further information on the organization/employees [11].

As a note, we can see how specific procedures of double-checking e.g. money transfers and information disclosures by calling the responsables or the sender and general vigilance of employees can hinder these attacks, which were some of the most repeated advice in the sources surveyed in Section 2.3.

⁶⁸The use of the target's national language or *top level domain* (TLD) can greatly improve credibility [7].

⁶⁹<https://www.tvsyd.dk/artikel/svindel-virksomhed-betalte-falsk-faktura-paa-100000-kroner>, <https://fashionforum.dk/2013/03/22/svindlere-gar-malrettet-efter-danske-modehuse/>, <http://lokalavisen.dk/hoejsaeson-for-fup-her-er-aarets-svindel-faktura-til-danske-virksomheder-fra-montenegro-20160830/artikler/308309988/1265> and <http://vafo.dk/erhverv/Vejle-firma-advarer-mod-falske-regninger/artikel/450316>

⁷⁰<https://www.b.dk/kultur/kriminelle-udgav-sig-for-at-vaere-direktoer-snoed-museum-for-805.000-kroner>

CEO-fraud/whaling Considered a specific kind of spear-phishing, this attack impersonates or targets (depending on sources⁷¹) C-level employees – also called *whales*, as they are “the big targets”.

The aim is to perform acts similar to spear-phishing, but due to the large amount of money that may be involved with C-level roles, a larger reward can be collected by the attacker. Prerequisites and traits of the attack are similar to spear-phishing as well; it may however **not be necessary to know any vendors/customers of the organization** for this attack, but only [27]:

- Name of head of the company.
- His e-mail address (to mimic or create something similar).
- Managers/employees authorized to perform a transfer of funds.

The emails may be even better crafted than regular spear-phishing emails through e.g. more formal/correct language [38].

Examples of *spear-phishing* can also be considered whaling; in a specific example, The National Museum of Art in Denmark were recently phished for 805.000 DKK⁷².

In-person/“physical” attacks If the attacker is willing to interact directly with the target/human sources of information in general by e.g. appearing physically on location or calling, an even wider range of scenarios are possible. These are naturally targeted in nature, as the attacker must choose some specific organization/place to appear physically.

Most of the scenarios described by Kevin Mitnick in [35] has some physical element to them. They built upon spear-phishing attacks, but requires human interaction, methodically planning and agility of the attack plan. The attacks depicted in [35] are diverse in their necessity of information required to work, but all exploit the human mind (i.e. *social engineering*) by different methods as described in Section 2.4.

As an example, the story named “*Not as safe as you think*”⁷³ describes how the attacker through human interaction by phone only acquires internal hostnames, credentials to these, out-of-office voice-mails, phone (with internal extensions) and fax numbers, dial-in access⁷⁴ and in the end, the data of some project. For this attack, the information found from OSINT-sources beforehand was only:

- Some personal data to verify with (date of birth, family info, social security number etc.).

⁷¹For example, <https://www.knowbe4.com/ceo-fraud> and <https://www.ncsc.gov.uk/guidance/whaling-how-it-works-and-what-your-organisation-can-do-about-it> states whaling is to target C-level employees, as they have the most privileges on the network, while e.g. [27], <https://krebsonsecurity.com/tag/ceo-fraud/> and <https://danskebank.dk/da-dk/Erhverv/Mellem-erhverv/Online-services/Sikkerhed/Pages/CEO-Fraud.aspx> states the attack is impersonation of bosses to convince an employee to transfer funds.

⁷²See <https://www.b.dk/kultur/kriminelle-udgav-sig-for-at-vaere-direktoer-snoed-museum-for-805.000-kroner>

⁷³Found on p. 64 of [35].

⁷⁴Probably what we will call a VPN today.

- Employees of different departments (only a few were necessary, the rest were offered by the employees he called⁷⁵).
- Company locations/sites

The rest of the information were discovered during the course of the attack. It should however be noted, that this story involves violation of many policies implemented in modern organization with controls such as those in DS/ISO 27001; those might however fail if the awareness among the employees are not sufficient.

Another method could be for the attacker to show up on premises, which requires proper attire of employees, vendors, shipping handlers etc., and maybe some knowledge of company behavior or locations; afterwards he can use social engineering-techniques to recover the necessary information.

Attacks of this type not requiring any particular OSINT-data includes baiting with infectious USB-devices dropped on the organization's parking/grounds, tailgating (following employees) inside the organization's premises or *dumpster diving* to recover confidential information.

Subverting the supply chain An example from [37], this is “*to attack equipment or software being delivered to the organisation*”. Its goal is to deliver a payload through the regular supply chain of the organization; suppliers which the organization has already put a high level of trust in and perhaps thus are less likely to question deliveries/content from.

We know NSA performs this practice against hardware/servers exported from the US⁷⁶ and some believe that Huawei-equipment⁷⁷ does the same⁷⁸. In a specific attack⁷⁹, a vendor of scanners running Microsoft XP Embedded OS were shipped with malware (named *Zombie Zero*). The malware targeted ERP-systems⁸⁰ of shipping and logistics and later also manufacturers.

It is not specified what information the attacker (which is unidentified and most likely an APT-group) had acquired beforehand, but we can make a qualified guess. A lot of information may have gone into compromising the manufacturer, but from the target organization, it may only necessary to know:

- A type of software used in the company (here an ERP-system).
- A type of hardware deployed (here a scanner) or the distributor bought from.
- The attack could also be leveraged by identifying the employee responsible of procurement of IT equipment (and contact information) e.g. from an organizational chart of social media.

⁷⁵The namedropping tactic is the primary driver behind the story “*Mr. Bigg wants this*” on p. 110 of [35] and in general attacks of the CEO-fraud-type.

⁷⁶<https://www.theguardian.com/books/2014/may/12/glenn-greenwald-nsa-tampers-us-internet-routers-snowden>

⁷⁷This is what is stated in the source article implying that maybe a state-sponsored group is behind.

⁷⁸<http://www.cbsnews.com/news/huawei-probed-for-security-espionage-risk/>

⁷⁹<https://www.forbes.com/sites/kurtmarko/2014/07/10/trojan-hardware-spreads-aps/#71924d852536>

⁸⁰*Enterprise resource planning* systems control areas like procurement, sales, economics and inventory control of organizations.

Targeted (D)DoS This attack can be employed if the attacker is politically motivated and wants to shut down a service/website/operations of the organization, but also as a tool of extortion. The danger of this is inherent in servers connected to the Internet; hostnames are quick to resolve and target, but if public IP's not meant to be exposed/used by regular users, are found through e.g. Shodan or public pastes of stolen data, the right security measures might not be present. It can also be a problem if internal IP-addresses are found from e.g. internal documents or in website descriptions, as it can be used to claim credibility (by proving knowledge of internal network components). Additionally, techniques exist to route traffic through public IP's to unintended servers inside the network.

Some of the OSINT that could be exposed online might even be of such a personal character that it enables brute force of login to e.g. web services. Arguably we could also consider adding a scenario for this, but have not done so for now, as it to some extent is covered by the spear-phishing scenario. It can however also just be plain "drive-by" attacks attempting login with credentials from public leak databases, where it thus will fit better in Section 2.5.2.

2.5.2 Un-targeted attacks

An un-targeted attack is when the attacker targets a broad range of vulnerable systems/human targets and do not care about if some specific organization/individual is hit. Information may be found using commodity search engines (Google, Shodan), tools (Maltego transforms or the ones from Sec. 2.1.2.2) and methods e.g. automated delivery of the attack.

Examples of attack scenarios of this type are:

Phishing Acting in an automated fashion, this attack type often aims to acquire data which can be directly exploited for e.g. financial gain, gaining access to common web services or to deliver a malicious payload. Examples in Danish context is passwords, (bank) account information or NemID-key cards [8] and recently, delivery of especially ransomware (see below).

Recent attacks on Danish Internet users have not relied on knowledge of information on them prior to the attack, but on mass-sending homonymous emails or SMS's⁸¹ equipped with names and/or logos of well-known companies and a sense of urgency in their content [8, 27]. The emails/SMS's have been used to deliver malicious payloads to get the aforementioned data or have just directly been asking for the data (e.g. for NemID-key cards).

Examples of campaigns are PostNord⁸², NemID⁸³, Apple⁸⁴ or NETS⁸⁵. [35] p. 93-104 contains several stories about phishing as well, pointing out the well-known signs of

⁸¹Phone calls/telecom fraud, sometimes called "vishing", are also known to be used, but are not as prevalent in Denmark currently.

⁸²http://www2.postdanmark.dk/iis-nyheder/vis_nyhed.asp?ID=1626&NyhedType=generelle&vis=arkiv&usr=&year=2015

⁸³https://www.nemid.nu/dk-da/pas_paa_dit_nemid/phishing/

⁸⁴<https://support.apple.com/da-dk/HT201679>

⁸⁵Handles payments in Denmark, Dankort and NemID among others. <https://www.nets.eu/dk-da/nyheder/Pages/Advarsel-mod-phishing-mails,-snyde-sms%E2%80%99er-og-fup-telefonopkald.aspx>

typosquatting domains, bad grammar etc.

Phishing may also happen by exploiting websites or content deliver networks (CDN's) (for e.g. ads) to serve phishing sites instead of the expected content or malicious payloads; this is called *water holing* (see below).

To improve credibility of the email/SMS's, the attackers may use:

- User database leaks, maliciously traded email lists with names⁸⁶ or similar to acquire pairs of valid email addresses and names.
- Organization-specific email domains, preferably where the naming scheme can be deduced from the examples of email addresses found online. Valid addresses can then be created from e.g. organizational chart, a list of employees or data from social media (LinkedIn, Facebook).
- Mobile phone numbers from e.g. social media, contact lists or document meta-data⁸⁷ to deliver SMS's.

Phishing attacks can also be used in connection with *water holing* as a mediator for e.g. tech support scams or fake lotteries/sweepstakes [27] (see below).

Water holing In this scenario, the attackers may copy or compromise a site or a CDN in order to deliver a malicious payload or phishing sites⁸⁸. The attack can fall into both categories of targeted and un-targeted attacks; whilst not necessarily targeted at a specific employee, the choice of website can be one used by both a few employees, entire companies, demographics or countries.

Besides being able to perform the exploitation/infection of the “water hole” and employing common techniques for tracking browsers (keymaps, language installed, add-ons used etc.) to use e.g. correct language, OSINT can be used to enhance effectiveness (largely dependent on the degree of targeting cf. the above):

- Hobbies of employees (found through social media, names cross-searched to communities or the use of company email addresses outside the company environment).
- Vendors or customers, including services used like web hosting [37]; may rely on additional social engineering attacks on the vendor/customer. If the attack is not targeted on a specific organization or its employees, the choice can be vulnerable sites for vendors of entire industries.
- Technical knowledge of organizational infrastructure (location, IP-range, software types/versions) and browsing habits [37].

⁸⁶If a user enters details for a competition online of dubious origin, their data may end up being sold as part of mailing lists. These are typically sold by geography or interests in the same way legitimate companies sell demographics data.

⁸⁷As this is un-targeted attacks, the information can e.g. be found through scanning tools as the ones described in Section 2.1.2.2.

⁸⁸*Microsoft tech support scams, fake lotteries/sweepstakes* and drive-by attacks with e.g. Flash exploits are regular examples hereof; see e.g. this write-up of one such ad-CDN serving both in a sophisticated manner: <https://blog.malwarebytes.com/cybercrime/2017/05/roughed-the-anti-ad-blocker-malvertiser/>

- In an automated fashion for purely un-targeted attacks: Top visited sites⁸⁹ versus some exploit of e.g. the `metasploit` framework to inject malicious code.

“Water holing” can be countered by e.g. sink holing the domain on the organization’s DNS server (or a higher up DNS if possible and applicable) [8] or using web proxies to scan web traffic.

Ransomware This attacks encrypts content on the victim’s computer and demands a ransom to decrypt the content; the goal is commonly financial gain.

It is usually delivered as the malicious payload of un-targeted phishing emails or SMS’s with content and methods to improve credibility as described above under *phishing attacks*. It is thus not an “attack type” in itself, but more a result of a successful deployment of OSINT and social engineering by an attacker.

Scanning Scanning is noteworthy as a major source of OSINT collection from the organization itself (most likely unintended) and used in the preliminary phase of many un-targeted OSINT-enabled attacks to choose targets. As with water holing, it can serve both as a targeted (to acquire additional information of a chosen target organization) or an un-targeted attacks (to find vulnerable infrastructure to exploit some known technique on, technical or not).

The goal is to map infrastructure by simple tools in the same way as the pen-tester does in a intrusive or non-intrusive fashion (see Sec. 2.1.2).

For the attacker to launch a successful scan against his target, he can benefit from knowing:

- Some online presence of the target organization, e.g. a web server. If the web server is hosted externally, scanning the web server may reveal redirects to or integration with in-house services or written references to other online infrastructure.
- Owners or hosts or some of the infrastructure, may also hold other parts of the infrastructure. DK-Hostmaster can be a source to locate these owners and administrators.
- Physical locations.
- Specific hard- /software used (by e.g. some vendor, method of connectivity or version number) that the attacker know to be exploitable.

⁸⁹E.g. measured by Alexa rank: <http://www.alexa.com/siteinfo>

Chapter 3

Analysis

This chapter analyses the contemporary methods, practices and theory of the previous chapter to understand how an organization is required to control or reduce its footprint and data flowing to OSINT-sources, how they can do it, and how they can test their compliance level in regards to this. Especially we want to show how an organization (and its cyber security contractors) could enhance their testing using the deliveries of this thesis (see deliveries in Section 1.1.1).

The chapter first reiterates what standards and procedures apply to an organization acting under Danish law and how this may be handled in practice.

Next Section 3.2 shows how implementation of standards and guidelines can be supported and verified by the organization and in particular through external consultancy (using the contemporary methods presented in Section 2.1).

Finally Section 3.3 and 3.4 details how the deliveries of this thesis can enhance testing procedures and reporting for the organization in accordance with existing standards.

3.1 Cyber security in a work environment

An organization acting in today's world is compelled to be conscious and active about their cyber security. They need to secure intellectual property (IP), infrastructure, business partners (suppliers, vendors) and customers as well as comply to applicable national legislation and standards¹. It is a concern to both the organization, stakeholders and society as a whole that cyber security is prioritized and managed competently and according to the organization's risk.

In Section 2.3 current legislation, standards and guidelines applicable to Danish organizations were presented by gathering and examining information from Agency of Digitisation, the DS/ISO 27001 and CFCS. The results were complimented by guidance from other large organs such as the UK's NCSC and NIST from the US.

¹From 1st of May 2018 a big leap forward is taken by imposing regulations on all personal identifiable data of EU-citizens processed and stored by organizations. It is called *The General Data Protection Regulation* (GDPR). With violations follows fines of up to € 20 million or 4 % of annual turnover. See Sec. 2.3.8 for a bit more on GDPR as well as some references.

In its daily operations, the person responsible for the organization's cyber security² is responsible for the organization to adhere to both legislative requirements and business requirements. This creates conflicting interests, where e.g. the business wants to collect, process and store as much data about its customers as possible (without consent), whereas the law prescribes to only collect strictly necessary, non-identifiable data and store it for a short duration. Anymore than that requires explicit consent.

Meanwhile, the facing threats are increasingly advanced and persistent, making it is an every-day task to deny and deter attackers (see e.g. [11]) while still enabling business. In Section 2.5 it was described how daily operations leave *footprints*, which are to be controlled by proper procedures and awareness in the organization (cf. Sec. 2.3).

The person responsible has to be able to navigate between these stakeholders and constantly assess the trade-off between risk, legal and business to implement the *proper* security.

3.2 The role of external security consultants

These are difficult tasks, and depending on the organization's strategy a part of the security function may be outsourced to security consultants. Under the current trend with too few security professionals it can be more viable to buy the manpower needed, depending on the size, specialization/expertise required and regularity of the task. Most tasks can be fulfilled by the professional security consultancies operating in Denmark today³. A security consultant may handle more specific tasks as opposed to the "all-rounders" of the internal security function, meaning that they may excel in pen-testing, but not be proficient in e.g. legal requirements of Danish organizations. The consultancy firms will seek to employ people excelling in each area to combine their knowledge towards the customer, but it will raise the price of their work.

One of the specific tasks security consultants solves for organizations, is to verify the security level within a given scope by performing a vulnerability scan or pen-test. The specifics of a pen-test is demonstrated in detail in Section 2.1⁴.

Separate findings of the pen-test can also be utilized as stand-alone results; for example the data gathered on the target organization in the initial phases, can demonstrate the extent of OSINT-data available on the organization. This is otherwise difficult to measure, but strongly recommended to avoid by CPNI, The Federal CIO Council and NCSC, and implicitly following the requirements for awareness in the organization (the user must be aware of the risk in sharing extensive information on the organization on e.g. public web sites).

²Alternatively an the internal group security function if the organization is large enough, takes the responsibility seriously, has funding and has been able to acquire some of the few, proficient people in the field.

³Examples of tasks can be implementing GDPR, performing risk analysis, assessing legal compliance, log storage and -processing, incident response, creating awareness campaigns or implementing and configuring technical security measures like firewalls, end-point protection etc.

⁴As stated in Section 2.1.4.1 a vulnerability scan instead aims to discover and rank common exploits that can result in a compromise, where the full pen-test aims to demonstrate the ability to breach the organization's security.

The first, important step of the pen-test is to gather and map intelligence on the organization. This is commonly done by crawling online data sources⁵. In particular the consultant will as part of this query OSINT-sources and employ a *case management tool* to store and organize his findings.

The data found can be hard to categorize as discussed in the introduction (Sec. 1). Most of this information are pieces of data which the organization do not consider confidential in itself, but on the other hand to not publish publicly either [35]. The internal security function or the consultant will have to readily be able to recognize the value of each piece of information to an attacker to be able to mitigate the finding. This is a task with no final conclusion.

The security consultant will have to manage time spent on each part of a pen-test and as such be limited by scope of contract and budget of the client organization (defined in the initial phases of the pen-testing process cf. Section 2.1.1). For the intelligence gathering-phase this can greatly limit the amount of time and effort spent on applying the many, specialized tools found in e.g. Kali Linux (see Sec. 2.1.2.2); the consultant must prioritize to the best of his ability, but simple click-to-run gathering on existing findings with output to the same platform will be fast.

The click-to-run feature is a strength of using Maltego as a case management-tool: By performing basic intelligence gathering using the transforms⁶, adding collected intelligence from other tools and being able to perform further searches on the data, the researcher can save valuable time. By being able to move freely in any direction he can explore all directions the data may lead, resulting in a better understanding of the scenarios made possible by the findings. He will also get an enhanced overview and categorization of the findings through the graph representation.

3.3 Maltego transforms for Danish OSINT-sources

Unfortunately no transforms exist for Danish OSINT-sources in Maltego despite Denmark being a digitally mature country⁷ and having a national strategy⁸ for publishing data to enable transparency and new business opportunities; this leads to the first deliverable of this thesis: A set of transforms to enable search of Danish OSINT-sources

It is important that the transforms fulfill the following requirements:

- Follow Paterva’s design guidelines of Maltego to seamless work with the existing transforms so users can use readily use them:
 - Useful to a wide audience.
 - Test the transforms (bad quality will lead to the transforms being removed from the “Transform Hub”).

⁵See Sec. 2.1.2

⁶See Appendix A.1 for a basic explanation of Maltego.

⁷Ranked 11 in the world on the *Networked Readiness Index 2016*: <http://reports.weforum.org/global-information-technology-report-2016/infographics-and-shareables/>

⁸Denmark has joined the G8 *Open Data Charter* as mentioned in the introduction.

- Error messages should be adequately verbose and set at the correct level.
 - Document the use of the transforms.
 - Fill out the meta-data when adding the transforms to the “Transform Hub”.
 - Use API-keys correctly.
 - Transforms should work out-of-the-box (or return correct error codes).
 - Use standard Maltego entities⁹ if possible and plan design of the in- and outputs. This is important to the overall usability of the transforms and treated separately in the design section (Sec. 4.1.1).
 - Name your transforms similarly to group them in the Maltego GUI.
 - Make the transforms free or with a trial.
 - Remember licensing information.
 - Support your users.
 - Name entities consistently and such that their relation to your transform is obvious.
- Work for OSINT-sources that are expected to add value and reduce the work-load of the intelligence gathering-phase of e.g. a pen-test. This should also encourage the pen-tester/consultant to download and use the transforms.
 - Not exist on the Maltego-platform currently.
 - Be supplied through the Maltego-platform for easy integration and guaranteed standardized work flows across multiple platforms.

The requirements are largely functional-only. However, to *follow design guidelines* is a mix of both functional and non-functional requirements, as they both describe how the transforms should function and how they should be programmed (e.g. error messages and licensing).

Similarly, other non-functional requirements are implied by the above: Paterva offers only `PHP` and `python` for programming to Maltego, resource management, scalability and performance are negligible in this context and managed by Maltego, and availability and similar is implied from following the design guidelines etc.

The transforms are going to be used as an integral part of the Maltego platform. Thus they need to adhere to the guidelines [44] given by Paterva (the developer); this ensures functioning, recognizable transforms and a direction for the design of the transforms. It is a descriptor of a solid quality of the transforms and desire to download and use them for the users. It is also a necessity if the transforms at a later time are to be advertised on the “Transform Hub” in Maltego (which is an advantage as it makes the transforms easier to acquire for the interested).

⁹These are the most basic entities always assumed to be present. They are initially added to Maltego by installing the “Paterva CTAS” transforms on the “Transform Hub”. They are also listed here: https://docs.paterva.com/en/entity-guide/standard_entities/.

As it can be seen the guidelines contains sound advice on designing the transforms and offering them afterwards; the concrete advice on setting them up for commercial use is not treated in this thesis, but should be referenced when maturing the transforms for commercial use in the future.

The transforms become *useful to a wide audience* when they perform actions that can relate to many different instances of intelligence gathering. The choice of OSINT-data sources to interface with is thus important to fulfill the requirements and necessary to know before designing and implementing.

As we know from Section 2.1, both the pen-tester and the attacker, whose mind he tries to mimic, work in an agile way choosing the next step based on current findings and experience. Choosing the best sources to make the transforms for by considering all the possible paths the gathering phase can take is impossible and will likely result in spending time on building transforms that may be valuable to one pen-tester but not the next one.

Instead OSINT-sources should be chosen based on the initial knowledge we can expect the pen-tester to have. The initial knowledge can be information given directly as part of the scope of the pen-test or something known by virtually all people, e.g. the website, the company name, its general/main contact details or the location of its headquarters. Transforms already exist for crawling website content and search engines for contact details or name, so no extra value is added by making a transform for this.

In Denmark we have a transparent top level domain registry (“DK Hostmaster”), where owner data is easy to access through the public API¹⁰. This is also the case for the national register of companies “Centralt virksomhedsregister” (CVR) which provides full transparency on company information, owners, finances as well as historic changes to these¹¹. The same national agency managing CVR also offers a public catalog of data, currently with 198 different datasets¹² of varying content (e.g. location of bike parking, public transport data, road markings) published primarily by large municipalities (Copenhagen and Aarhus); Copenhagen municipality themselves currently offers 237 datasets¹³. This data could provide insight in a specialized pen-test but does not fulfill the requirement of a transform to have a wide audience. Similar registries exists for e.g. the national registry on property, “Bygnings- og Boligregistret” (BBR), the public registry of rights on e.g. real estate, other housing, cars, marriage contracts and personal property, “Tinglysningsretten” and the registry of cars “Motorregistret”.

Much of the data from BBR and Tinglysningsretten is gathered and available on OIS.dk (“Den offentlige informationsserver”) and can be accessed through API’s offered by commercial partners¹⁴. Similarly, data on cars from Tinglysningsretten, SKAT (the national agency of taxation) and Motorregistret is offered on a commercial basis from other actors.

¹⁰<https://github.com/DK-Hostmaster/whois-rest-service-specification>

¹¹See <http://datahub.virk.dk/dataset/system-til-system-adgang-til-cvr-data>. Use requires sign-up.

¹²See <http://datahub.virk.dk/data/search>

¹³See <http://data.kk.dk/>

¹⁴See <https://ois.dk/UI/0mOIS/0mOis.aspx>

The data found in all of the above registries are relevant to include in the intelligence gathering of a pen-test. Look-up in them only requires information known from the start of the engagement, thus fulfilling the requirement of being sources that are expected to add value in any intelligence gathering on any Danish target. Such transforms are not pre-existent on the platform either. It has also been considered if the standards and guidelines in Section 2.3 could be used to select information sources to write transforms for but no pointers were found there. Other Danish OSINT-data sources exist, but the above-mentioned can provide the required data for our transforms to follow the requirements and can thus be considered to design for.

3.4 An auto-generated OSINT report

Having performed the remaining phases of the pen-test, the consultant has to deliver a report on his work; this is described in Section 2.1.6. As mentioned in Section 3.2, the consultant can utilize the work from the intelligence gathering-phase to report on the amount of OSINT-data found on the organization which is an important but difficult part of an organization's security function to deny and deter attackers. Being able to deliver an additional report on an area difficult to comprehend and have a full overview of also leads to an increased value of the consultant's service which in turn allows for a higher pricing.

To be able to deliver this report, the consultant needs to detail the findings in an easily digestible overview. The internal security personnel may not be as technical knowledgeable as the consultant, so it should be clear *how* the findings relate to attack scenarios as well as requirements they may act under; this will also enable the use of the report upwards in the organization 'as-is', which internal security may find valuable to spread awareness on the management floor, enabling future funding. Similarly the consultant may have gained a some understanding of the possible attack vectors of the adversary as the pen-test will involve following the same patterns, but relating the results beyond the scenarios to current regulation and guidelines can prove more difficult. The consultant will supposedly find that being able to connect findings of the intelligence gathering to attack scenarios and regulation and guidelines requires a great overview of the data, outlook to real-life applications and is time-consuming.

To solve the problem of connecting the findings with plausible scenarios and applicable standards and guidelines and to conclude on the intelligence gathering-phase of a pen-test, a framework to auto-generate a standardized report from Maltego is proposed as the second delivery. In particular the auto-generated report should fulfill the following requirements:

- The report must contain a basic statistic summary categorizing the findings into different types of data, so that the organization will be able to identify areas in need for mitigative steps.
- The report must relate to common cyber attack scenarios. The attackers are dynamic and opportunistic, so there are no exact way to say that "this data found, will lead to

vulnerability to this attack” as with a software vulnerability. Instead we must show that some piece of OSINT-data found can be exploited for some kind of common type of attack so the client can understand the findings’ potential to enhance the attack (i.e. mediating awareness) and be able to act accordingly.

- The report must relate to applicable Danish guidelines and standards. This ensures relevancy to the client organization. By relating the findings to concrete guidelines/standards¹⁵ the value of the report is raised as it provides input on an area of cyber security that can be difficult to review the compliance of.
- The report should be easy to interpret by using simple figures.
The goal is to deliver maximum value to the customer organization; maximum value is to get the most important information in an easily digestible way, which is why reports often contain an *executive summary* that can convey the necessary information of the results of the report in only a couple of minutes.
The findings and the conclusions made in the report should also be easy to follow for the user such that he understands *why* and can accept or reject the conclusions at his own discretion in accordance with other results of his investigation; he should not deliver it to the client blindly.
- The report generation should be easy to perform. I.e. input should not need to be entered by hand, the user only need to make simple choices during generation and output should be usable “as-is” format and run on across most platforms.
- The report should work with the types of data that can be expected to be found with Maltego (or manually input hereto), so it can be readily generated in an automated fashion from the findings which are already put in a common data format as it resides on the Maltego platform.

Most of these requirements are functional, while arguably the two latter are primarily non-functional, as they describe how the program should run and technical requirements (e.g. it should work on output from Maltego), which in turn is a result of the functional requirement that the program should be usable to most security researchers, which we pointed out use Maltego for their work.

It is not necessary to specify e.g. availability as a requirement because we are not designing e.g. an entire production system.

Note that even though we primarily speak of consultants performing a full test of some organization here, the report is also relevant to e.g. an internal security function, a consultant with a small 2 hour task as well as others, that might have a need for simple insight on data available on OSINT-sources about some target.

¹⁵It were found that little to none of the legislation are applicable in this context due to high-order descriptions only.

The last bullet supports this, as these scenarios require the functionality of the report to be present after spending 10-15 minutes of searching.

Demonstration and choices of “common OSINT-enabled cyber attacks” is a subjective task; our proposition of these were given in Section 2.5. Here we identified eight different common scenarios based on a generalization of the attack phases of a social engineering-attack (Sec. 2.4.1) and the techniques and tricks used (Sec. 2.4.2). The scenarios do have a similarities which is unavoidable, as they built upon the same techniques of “cheating” people and OSINT-data, but some notable differences exist between each of them in regards to execution, methods or target. They are all commonly occurring in the current threat landscape and thus recognizable to the reader.

Each scenario is presented with categories of OSINT-data enabling them. It is these we can use to connect findings from an intelligence gathering-phase conducted in Maltego. We cannot include the “just asking”-scenarios or scenarios based solely on social engineer-techniques (e.g. acts of sympathy/guilt/intimidation), as they cannot be related to the actual findings.

Structuring the scenarios are different from what we normally see in threat modeling (cf. Sec. 2.1.3). We have focused on different methods of exploitation as e.g. [35] does, while the norm is to focus on the attacker’s goals. It is not a viable option here because the “attacker” is an unidentified entity with varying goals. Grouping by method also contributes by offering some familiarity with scenario-types often referenced in the news. In turn it improves the recognizability and correspondance the scenarios in the auto-generated report. It is important however to disclaim to the reader that the social engineer has several other different option-/weapons in his arsenal and that “no fulfilled scenarios” does not equal resilience towards social engineering as a whole (e.g. deceiving, questioning employees, the “just asking”-scenarios or acts of sympathy/guilt/intimidation).

To produce the added value of the report, we want to link findings with the exact legislation, controls of standards and guidelines applying to Danish organizations.

To identify *all* the relevant legal requirements, standards and guidance available to Danish organization is a difficult task when one is not previously well-acquainted with this from e.g. daily work; the task of getting the full picture, is out of scope for this thesis and these studies. Section 2.3 seeks to put forward the relevant content by examining government bodies that have presented resources to provide this overview. In a Danish context we find Agency for Digitisation, CFCS and the widely used DS/ISO 27000-series; this is complemented by UK and US bodies (CPNI, NCSC, NIST and similar). We can not expect to have a complete, exhaustive list, as a lot of guidance exists; security professionals may also have a differentiating experience of what is most important – again pointing to how the field is complex. The three sources are however prominent in Danish context and are complemented by guidance from other large bodies.

In Section 2.3 it is found that current legislation cannot be implemented in a report in its current

form. Most of it is not focused on *outbound sharing* (as we learned in Sec. 2.3.6) and the one that is, is kept in general terms to use for controls in the organization, where we would be required to interpret it to our cases or acquire concrete implementation guidance.

Nonetheless, the section do present content (concrete standards/guidelines) that *is* relevant and we seek to connect the content that can be linked with specific findings and present them as a list of “violations” that the client organization can consult.

It is important for both the security consultant and the client to be presented with a disclaimer so they are aware of the limitations of the current iteration of the auto-generated report as it is not guaranteed to provide a full check-list of compliance with the sources of Section 2.3 in the same way that the e.g. controls of DS/ISO 27001 can be “ticked off” during an audit.

The summary of findings in the report should focus on being easy-read. We saw from the examples in Section 2.1.6 how this can be done with simple colors (in particular used in the Dubex SAC-report found in Appendix B.2) and frequency maps (in particular used in the Qualys-report found in Appendix B.1). The summary should especially convey the frequency of the findings in general and comparatively between them. This data can be well presented in tables making it easy to read off values¹⁶, but to make it easier to display trends bar graphs are also a good choice¹⁷.

If it is required to demonstrate a difference in severity between some data, coloring can be used as in the Dubex SAC-report to indicate this; in turn, the coloring can be used to summarize the report in relation to the written executive summary of the report (which are written and delivered with the report by the consultant).

It is necessary to list the data off which the conclusion in the executive summary is based. The coupling of findings with legislation and guidelines are as mentioned arguably differing between professionals so the reader has to be able to follow the trail and investigate if it fits his conclusions. This can be done in subsequent sections after the primary conclusive sections listing the data satisfying the requirements of each scenario or guideline. Also here colors can be used to make the report easier to interpret.

The final design of the report is decided in Section 4.2.1.

Finally, the report should be integrated or usable with an existing platform. This covers implementation-specific questions otherwise belonging in Section 4.2.3, but has to be discussed here, at it justifies the use of Maltego-data as input moving forward.

We learned from Section 2.1.2 that the researcher is likely to use a case management tool in this phase and we found that Maltego provided a platform for both this and the transforms. Therefore it is preferable to base the input for the auto-generated report on an export of Maltego findings.

Maltego offers two ways of export from the GUI:

- A csv-file with rows consisting of the value of a parent entity and one child entity; there

¹⁶<http://web.mit.edu/course/21/21.guide/tables.htm>

¹⁷According to <http://web.mit.edu/course/21/21.guide/grf-bar.htm>

are no info of entity types nor additional properties of the entities (see an example hereof in App. D.2).

- A **pdf**-file containing the full graph, three top 10-lists of entities ranked by *incoming links*, *outgoing links* and *total links*, and all entities listed by type. As with the **csv**-file, additional properties of the entities are not included.

Another possibility might exist: To perform transforms, Maltego uploads the value or the properties of the entity, on which the transform is launched, in a non-compliant XML-format¹⁸, which is parsed to the transform code on the transform host server.

Unfortunately, the parsed values does not contain the additional properties here either. If it were to be used, it would require some editing of both the Maltego-parser as well as “hacking” the transform code to combine input from several transforms into one and output to disk. It may be a viable option, but depends on further scrutiny and an unnecessary risk of dead-ending this thesis-delivery.

Neither of these options are desirable for automatically generating a report. Only the **csv**-file can be used directly and will still require some user input to categorize the data. Additional properties of entities are not included in the export.

If we instead look at tools whose main purpose is the collection of information, this would still entail some (manual) pre-processing, because no tools exist to categorize the information and the value of each piece of information is dependent on the context in which it occurs. Alas, to this end the Maltego-platform is very limiting. We have to rely on the **csv**-file for inputs and will need the user to categorize and enrich inputs. The Maltego-generated **pdf**-file can be used in combination with the data we process to show frequencies of data depending on the specific program design.

In all cases, it will need the researcher to remove any unnecessary data from the graph before exporting the data to avoid wrong statistics in the Maltego-generated **pdf** and many entries to be discarded from the user categorization.

3.4.1 Categorizing the data for the auto-generated report

To automatically relate the categories of information, the scenarios and the guidance, we need to have structure of linking them to later design the programme and its data structure from.

The mindmap in Figure 3.1 presents the content that we need to relate the report to (as per Sec. 3.4) to give value to it. The content is based on Section 2.3 for all relevant standards and guidance pertaining to the findings as well as Section 2.5 for common attacks exploiting the findings made. Naming and numbering is used to allow for reference to this thesis report (both scenarios and standards) or the individual guides.

The mindmap contains only policies/rules/guidance that are applicable to concrete findings.

¹⁸Determined by examining the parser supplied with the code for the transform host server used as part of this project.

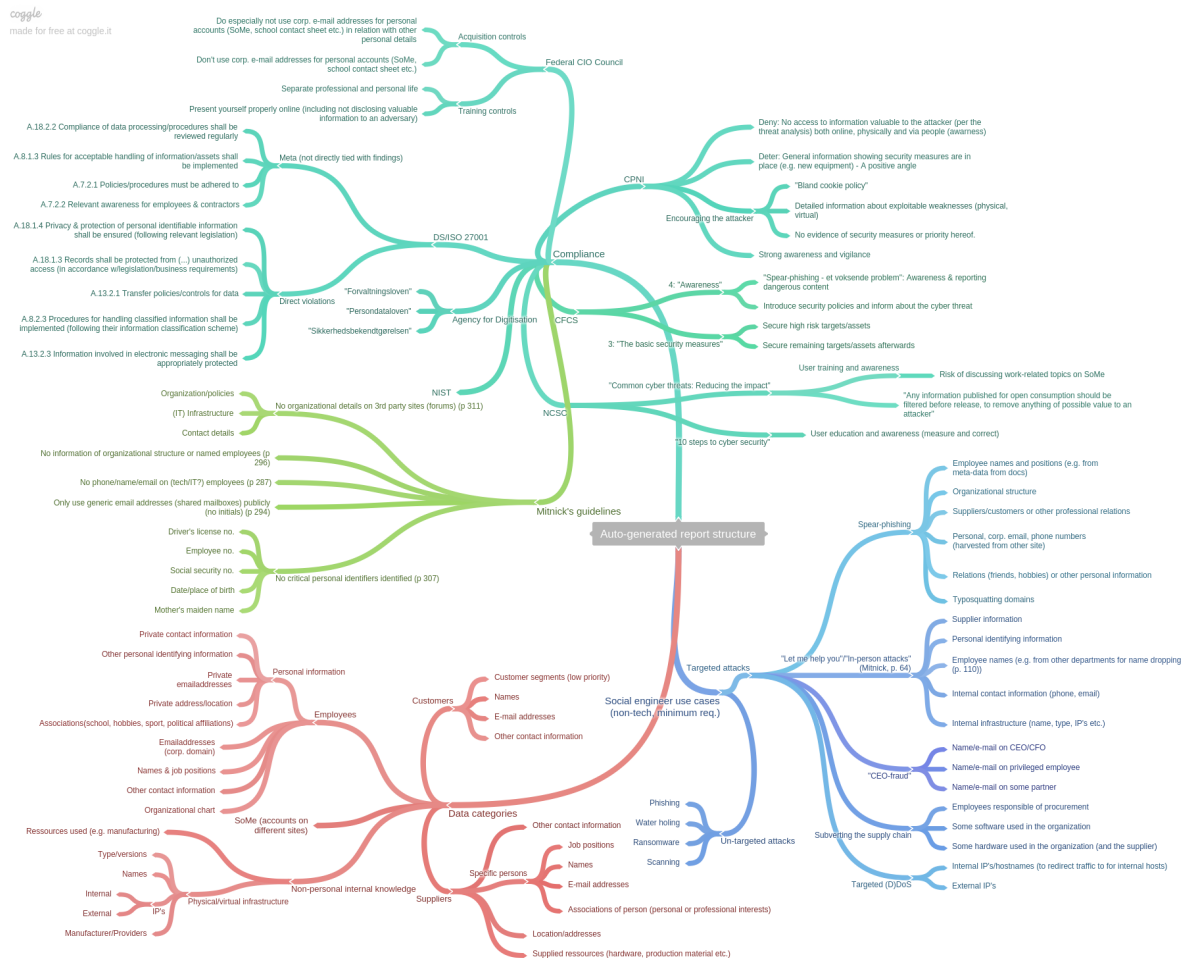


Figure 3.1: The mindmap for the content of the auto-generated report. A structure for the un-targeted common attack scenarios of Sec. 2.5 are not currently displayed, but will follow that of the targeted. A larger version can be seen on Fig. D.1.

From the sources of Section 2.3, we picked out relevant “requirements” or elements of them to use, which are possible to link with findings. With further consideration, only the policies on the branches pertaining to [35, 25, 15] are implementable with the current data categories (see below for extension and alternative approaches that may encompass other standards/guidance). The mindmap is organized such that the branch “data categories” contains the labels of which one or more are expected to be put on each findings. Each label is linked to none, one or more requirements of scenarios and guidelines, so the presence of a label can be linked to satisfaction of requirements of scenarios/guidance to later be outputted in the report.

While a selection has been made in the content from some of the sources in Section 2.3 (see e.g. the controls of the DS/ISO 27001 (Sec. 2.3.1.1) which are so plenty, we could not describe them all), other sources were presented with policies that could be used here, but needed the context of the source’s advice in its entirety to make sense. This means that some controls (e.g. Table 2.6 on mitigative controls per attack stage) or advice are left out from the mindmap, as

they would not represent a control that could be related to any findings of the report, thus only confusing the picture.

With CFCS the content cannot be directly tied with actual findings. It is merely advice that we potentially can tie with some categories of findings in general, but still there will no direct proof that the advice is “violated” just because some data within a specific category is identified.

Specifically in step 4 in [20] awareness is mentioned, which we cannot measure using Maltego transforms, so we cannot expect that information to be in the data export from Maltego. We can only measure that an employee has not been aware of the threat by sharing some specific piece of information. This do allow us to point out the relevancy of reviewing the procedures surrounding the awareness programme of the organization, but it is necessary to know the context in which the data was found. Maltego has a very limited output, so we cannot capture this.

It is a difficult task finding the perfect set of requirements of scenarios and guidelines to combine with some information categories – even the term “perfect” is quite subjective due to differing opinions and perceptions within the field; it is probably a contributing reason why such tool do not already exist.

The standards and attack scenarios comes from two different domains; there is a huge difference in perspective and granularity with the scenarios aiming to explain events and the standards policies, controls and managerial course of actions. The data categories are made from a software engineer’s perspective and atomized by subject within the field, which do not necessarily match the two others (standards and attack scenarios).

An improved version requires further work on making all three categories form one, coherent mesh of interrelated data categories and requirements. In particular, bringing in professionals to give input (especially with varying backgrounds) would be valuable.

Our design uses an 1-to-1 correspondence between a finding and a requirement being satisfied. The standards/guidelines often rely on knowing the circumstances the data appeared under (e.g. found on the organization’s website or a 3rd party). We have not implemented this currently and have not connected data categories to standards, were this is a necessity.

An improved approach could be to require several findings within each requirement, before considering it satisfied. This allows for differentiating between *strong* and *weakly* satisfied requirements or requirements needing several findings to form a viable attack (which is possible to do for some of the scenarios).

Another choice (not mutual exclusive from the first) is to not only distinguish between labels by categories (which easily become quite atomized resulting in a mass of labels, difficult to differentiate between to both the user and the designer), but rather have fewer categories, closer related to scenarios and guidance and be able to mark the specific finding “valuable” in the context of the applied label. This allows for some scenarios only using the valuable findings (e.g. the “CEO-fraud” scenario) and other scenarios using the same label for requirements with a lesser prerequisite. Additionally it may allow for other, broader categories, which in turn can enable links with the “softer” controls of the standards/guidance.

The current state do however demonstrate the viability of this approach: It is possible to automatically input data and link them with scenarios and guidelines, but it happens within the constraint that the links are highly subjective! To this end we have not succeeded mapping the links, because they are currently only based on the author's immediate understanding of what requirements make up each scenario and guideline. The approach is 1-to-1 correspondence based on the data labels, but the requirements are in several cases not linkable to the labels, because the granularity and perspective is different.

The current flow of selecting labels are manageable in number and has clear primary categories, so the analyst are expected to be able to distinguish between. For real-life application it would be advisable to include documentation on the labels and use of the frameworks as a whole.

Chapter 4

Design & implementation

This chapter describes the design and specific implementation of the two deliveries (see Sec. 1.1.1) based on the requirements developed in Chapter 3. Both deliveries are to be programmed and interfaced with the Maltego-platform, so the chapter describes considerations and concrete coding necessary to fulfill this and the requirements set up.

The chapter treats the design and implementation of the Maltego transforms and the software to auto-generate the report separately, as they are different in setup, I/O etc.:

- In Section 4.1 we design and implement the transforms for Maltego, specifically transforms for DK Hostmaster in Section 4.1.2 and for the register of Danish vehicles (and related data) in Section 4.1.3. The implementation is described in Section 4.1.
- In Section 4.2 we look at the auto-generated report, specifically the design in Section 4.2.1 and the implementation of it in Section 4.2.3.

4.1 Maltego transforms

Designing the Maltego transforms is largely shaped by the Maltego platform. By running there, the transforms are ensured properties as availability, performance etc. which are tied to the Paterva environment. In order to have the transforms run on the platform, we need to adhere to the design guidelines [44] discussed in Section 3.3.

The general flow of data in a transform is depicted by Paterva in Figure A.1 and a bit more accurate (for our application) in Figure 4.1 (sequence diagrams are optimal for showing interactions [22]). Execution will always return data to the Maltego GUI. In case of errors, these are shown to the user. They can both be errors from the data source, the transform code and the transform servers, which highlights the need for informative error messages. These are returned with a built-in method that can be used to `try-catch` errors and return the message to the output window in Maltego; the transform can still return a partial resultset. If errors are not caught however, a pop-up containing the error is returned and no resultset. Additionally we can output

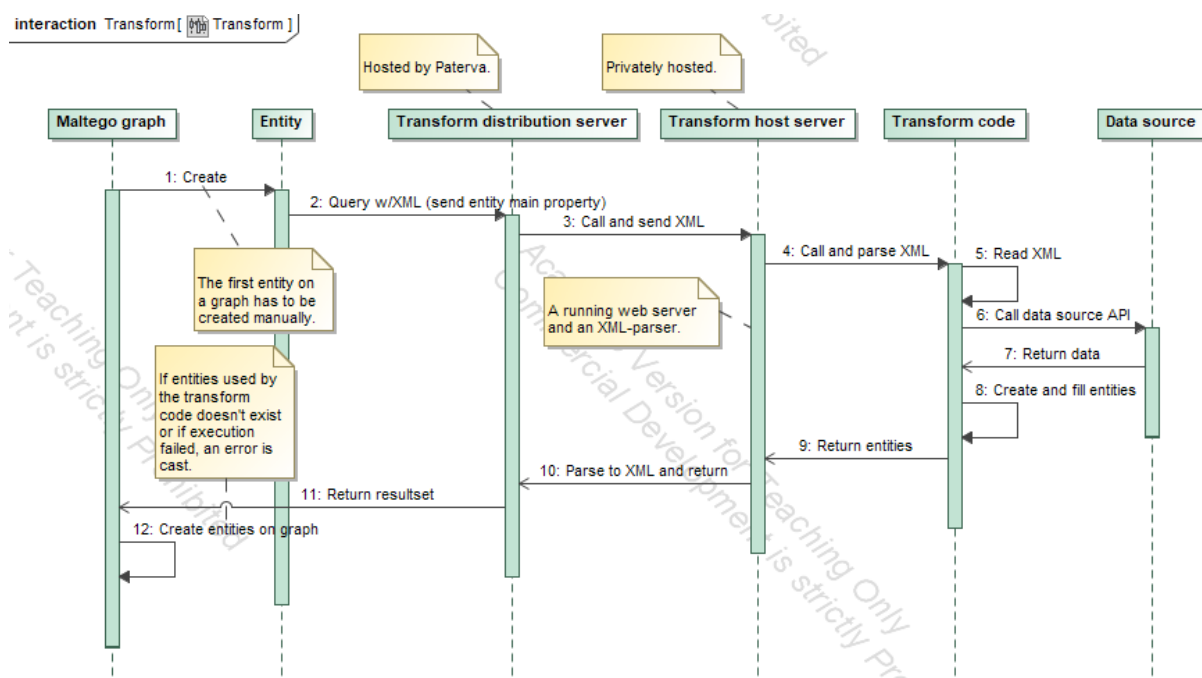


Figure 4.1: Sequence diagram of the dataflow in a custom transform querying a data source.

relevant information using the `UIMessage`-function in the Maltego-development framework. Care should be taken not to expose information about the API though, as it may not be publicly accessible (e.g. API URL or -key).

In Section 5.1 we show how the transform’s function are tested.

It is beyond the scope of this thesis to develop transforms covering all possible/usable sources to gather data from. It would however certainly be ideal to continue development for additional sources at a later time, as there is a potential for a minor capitalization of the transforms to Danish security consultancies.

A prioritization of which sources to develop for can be done depending on documentation of the API’s, data quality, “usefulness”, pricing (if any) and general availability (up-time, allowed amount of queries). To this end, DK Hostmaster’s and CVR’s API’s are offered fully public, whereas data from OIS.dk requires contact with the commercial partners. The data on cars (e.g. look-up on number plates) are offered by a couple of commercial partners, whom themselves have created a business by aggregating and cleaning data from public sources (as intended by offering open data)¹

DK Hostmaster and CVR are good sources to start from. They are useful to a wide audience (with Danish interests) because all `.dk`-domains are handled by DK Hostmaster, whom provides extensive information and a identity (a “handle”) used for each account across the database². Similarly, all companies operating in Denmark has a CVR-number and CVR enables look-up of those and a public, well-documented API³. Documentation is plenty, so the development will be less

¹Examples hereof are <http://nummerplade.net/> and <http://nrpla.de/>.

²They offer this through a public REST API: <https://github.com/DK-Hostmaster/whois-rest-service-specification>.

³See <http://datahub.virk.dk/dataset/system-til-system-adgang-til-cvr-data>. Use requires sign-up.

likely to be caught waiting for answer from some support and there will be no cost to use the API. Both providers are well-established (basis for high up-time) and offer unlimited amount of queries.

However to this end, we have chosen to develop a proof-of-concept only for DK Hostmaster and the aggregated data on cars offered by <http://nrpla.de/> respectively⁴

Both sources contain data about a lot of organizations/individuals in Denmark: DK Hostmaster offers information of registrant and administrator (name, address, account name (a “handle”) and account type) for all .dk-domains, while the data on nrpla.de/ contains all data from the car register, debt (from the public notar “Tinglysningsretten”; including names of both creditors and debtors), insurance and inspections, both current and historical. Thus they can both demonstrate the usefulness to a security researcher’s work across different cases. The current WHOIS-transforms in Maltego does not support .dk-domains, and while the combined data on cars are publicly available in similar fashion as DK Hostmaster provides, no-one has developed transforms for either source.

4.1.1 Designing entities

An important part in the Maltego design guidelines [44] is guiding the entities used for in- and output of a transform (separately treated in [43]).

Searching in Maltego is largely driven by transforms, which are available per entity such that one entity has one or more transform(s) that can be executed on them.; it is only the main property (typically the value displayed on the graph) that is parsed to the transform. The analysis can take many directions and a lot of entities can be used by the analyst (both originating from other transforms and manual input).

To not clutter the view or confuse the analyst about which entity to choose and continue further search, a minimum of additional, custom entities are preferable. This also enables our transforms to be presented together with existing transforms in the GUI when users use existing entities and use transforms on these.

Paterva is adamant that developed transforms and entities supports continuous search and do not leave “dead-end entities” (entities with no transforms for) [43].

We avoid the dead-ends by aiming to primarily return Maltego-provided entities (i.e. those in the group of “standard” entity group). The user still needs to add these to a fresh install of Maltego, but seeing as Maltego virtually cannot be used without them, it is a valid assumption that they are present on the installation when the transform is executed.

In case custom entities are developed, they should either inherit from a Maltego-entity (in that way transforms for the Maltego-entity also works for the custom entity) or other transforms should use them as input. This enhances reproducibility and ease of use, as the analyst does not have to convert to special entities. When creating the custom entities with inheritance, we get all existing properties of the parent entity, but the option to freely choose the main- (the one

⁴Developing for CVR was not an option due to a quite prolonged process of acquiring access taking well over two months without credentials being provided to me.

parsed to a transform) and display name (the one shown in a graph) and create new properties. Paterva use default values for some of these properties, which does not behave as intended⁵, which does impose some restrictions on the actual design on the entities in this project (discussed later; see also App. A.2.1 for a detailed description and Paterva’s work-around).

Only using standard entities, some specific information can be hard to display properly and must be added at run-time in dynamic fields, as note or as a *phrase*-entity (essentially only just a **string**, but as an entity), which does not convey as much information about the data piece as a custom entity may do, hence their usefulness.

For specific guidance on Maltego-development, refer to Appendix A. In particular distribution of the developed transforms are shown in Appendix A.2.2.

4.1.2 DK Hostmaster-transforms

In this section we detail how the transforms interacting with DK Hostmaster are designed and implemented.

DK Hostmaster has the full registry of all registrars, owners and nameservers of **.dk**-domains with name, address and contact information for the two former. All this data are referenced by a “handle”, which is the unique descriptor of the accounts. One entity can have several handles/accounts, but a handle can only point to one entity. The handles can be either personal, private companies, public organizations or an association; the transform shall be able to return data on all of these.

DK Hostmaster’s API offers look-ups for domains, handles and nameservers. The API also lists a method for *domain lists*, but currently (July 2017) this entry is unfinished in the documentation. All three methods are relevant to code transforms for, as they fulfill the requirement of *being useful to a wide audience*. As the look-up of nameservers are also contained within regular domains, only two are needed.

For both transforms, we assert that the resultset from DK Hostmaster is following the documentation, that is it only returns a **HTTP 200**-message if the resultset is correct. Thus we only have to handle errors on the input.

4.1.2.1 Domain-transform

As input for this transform, we can rely on the pre-existing entity `maltego.Domain` and return an error, if the domain is not a **.dk** TLD. A researcher will be very likely to use this entity if he wants to investigate a domain.

The output from the API is extensive. The method returns full information on the domain’s admin and the registrant (including handle), creation and expiration date and nameservers

⁵Confirmed through extensive e-mail communication with their support.

(including handle)⁶.

From the results returned by the API, we determine that the relevant information to return to the user is:

- Administrator & registrant (name, contact info, address, handle, account type)
- Expiration date (“paid until-date”)
- Nameserver(s)
- Nameserver(s) contact (handle)

The only excluded information is the use of DNSSEC and the creation date, which is less relevant to know, but could easily be implemented. Not all information is present on all domains, which has to be taken into account when coding.

The above information can be contained in the following entities, of which only three are custom, but all inherits from standard entities. They are named in an identifiable manner as required by [44].

For all returned entities, we mark the origin of the data in them (domain, handle, role) appropriately with the `addDisplayInformation`-method for easy reference. If two entities are identical, the display information is also joined, so it is still evident that it originates from two origins.

thesis.dkhostmaster.handle Inheriting from `maltego.Phrase` this entity contains the handle of administrator, registrant or nameserver contact as main property and handle and account type (if found present).

thesis.location An address; inheriting from `maltego.Location` with appropriately added fields for what DK Hostmaster returns (zip code, street address line 2 and 3 and a “attention”-field). The displayed property is the address, zip and city of the handle, but the main property is the city and country (due to an error in Maltego explained below).

thesis.expiry A date⁷ showing when the domain expires; inheriting from `maltego.Phrase`.

maltego.PhoneNumber To display landline/fax numbers (if present).

maltego.PhoneNumberMobile To display mobile phone number (if present).

maltego.DNSName Contains the hostname of the nameserver(s).

maltego.Person From determining the account type (if it is *Personal*), we can decide on returning the name of the admin/registrator as a separate entity to enable further search of the individual associated with the account.

⁶See <https://github.com/DK-Hostmaster/whois-rest-service-specification#domain>

⁷Surprisingly not an entity existing in Maltego!

As all three custom entities inherits from existing entities, we avoid “dead-ends”. The handle-entity can even be used as input to the handle-transform. We only create the strictly necessary entities as [44] endorses, but we still extract all useful information from the resultset and present it appropriately.

We note that the naming of the custom entities are in line with the requirements of the Maltego guidelines.

As the main property is used to run further transforms on an entity, we put the primary important information here. Only the two first entities have several properties, were the main property has to be selected appropriately to ensure further search.

The only entity where design could have been done differently is `thesis.location`, where a Maltego bug meant we were forced to use a concatenation of city and country as the main property. This is not optimal for either future search or distinguishing entities.

In Danish context we would normally use the street address, the zip code and the city to uniquely identify an address meaning we would put that into the main property.

However in `maltego.Location` Paterva overwrites the main property with the contents of the city- and country-property, which developers cannot change. Alas, when using a custom entity, we only have the option to choose either the same main property the parent entity has or properties of the custom entity, so we have to create a property only for the display purposes; in this we can concatenate street address, zip and city as desired.

This was one of the major problems found during development; it is explained further in Section 5.1⁸.

We could also have chosen to use the name associated with the handle as display value. It does not make a big difference as all data are still present in separate properties for the researcher to extract himself, but the street, zip and city conveys more information at a glance than the name only.

4.1.2.2 Handle-transform

Input for this transform can be done either with a generic entity like `maltego.Phrase` or a custom entity. The guidelines dictates not to create too many custom entities, but the handles are used a primary identifier by DK Hostmaster and as output from the domain-transform, so it does make sense to use as input. We return an error if the handle does not exist.

The method returns all information on a given handle⁹, but not a list of domains it administers (the method `domain list` should perform this, but its entry is unfinished).

It would be nice to instead add information to the input-entity, but Maltego does not make that possible.

The relevant information to return to the user from this API-method is:

⁸See also an excerpt from the e-mail conversation in App. A.2.1.

⁹See <https://github.com/DK-Hostmaster/whois-rest-service-specification#handle>

- Handle name, contact info, address, account type

We contain this information in the following entities, where only one is custom (and inherits from a Maltego entity):

thesis.location As for the domain-transform, contains address of the handle. The displayed property is the address, zip and city of the handle, the main property is the city and country (as explained above).

maltego.PhoneNumber To display landline/fax numbers (if present).

maltego.PhoneNumberMobile To display mobile phone number (if present).

maltego.Person The name of the admin/registrator (only if the account type is **Personal**).

maltego.Phrase The type of the account behind the handle (if not a **Personal**).

The only difference from the entities of the domain-transform is that we add the account type as a separate entity. For a personal account, we add can add the name of the account as a person, which it enables a whole other group of transforms to be used to search for individuals. If the account is not personal, we just note this in a **phrase**-entity, because it is an important piece of information, but it does not belong in any of the other entities or as a custom entity.

4.1.3 License plate-transforms

The API from `nrpla.de` offers a unified API for accessing the data they have aggregated from SKAT, DMR, Tinglysning and on inspections¹⁰. The entrance is a *vehicle identification number* (VIN)¹¹ or a license plate. This returns some basic information on the car and a *vehicle ID*, which can be used for searching in the other registers.

The API is protected with a token, but a sample output can be seen in Appendix D.1.

As can be seen from the sample output, currently no other transforms exist for most of the data coming out from the API of `nrpla.de`, so to follow Paterva's guidelines and avoid "dead-ends" all the car data goes into one entity because it cannot be used in further transforms; the car entity in itself will however be a dead-end, but that is unavoidable.

From the inspections and debt information we do get locations, names and CVR-numbers, which should be treated the same way as in the DK Hostmaster transforms. As the CVR-transform are not implemented for now, we do however skip that.

As input for the transform, the API offers both VIN and a license plate. Both are less common to find during an investigation and we can assume that no other transforms provide this as an output. We can thus either choose to create entities just for input to this transform or rely on

¹⁰I think this originates from one of the other sources after getting in contact with some people at SKAT who knew of the inner workings, but I'm not sure.

¹¹*Stelnummer*

`maltego.Phrase` as input (which already is a catch-all miscellaneous entity as mentioned). For now, we move forward with specific entities for the input, as it highlights that the possibility exists for the user to input and run transform on this data. Developing transforms not existing prior in the tool will need some way to nudge the user to employ them in his routines and this design can aid that.

We will create a `thesis.VIN` and `thesis.licenseplate` entity. In Maltego context two separate transforms needs to be created for that, but the code behind is identical and just looks at the entity input type when deciding which URL to query.

The result set as presented in Appendix D.1 gives rise to the following entities in the output:

thesis.car Contains virtually all information that can be deemed relevant for a researcher to determine enhance social engineering-enabled attacks: Details about the car, taxes, insurance (including which company), mileage, debt, debtors/creditors (if available) and leasing (if applicable). Has the brand, model and version as display- and main property. All fields from the result implemented here, are marked in the table in Appendix D.1.

thesis.location Outputs data on inspections separately by each inspection hall, complete with any remarks (the specific remarks will not be implemented for now). In this data set, the only way to establish approximate location of the car. Uses the name of the inspection hall as displayed property.

maltego.Person If any persons are found in the data, this is output here; this can only be found, when Tinglysning has registered debt in the car and is the only way to link a car with a person in the data set.

In a later iteration it will make sense for the insurance company and the inspection halls to be output to entities per CVR-number such that it is possible to further search on these, should that be desired. With the current transforms available, this does however not comply with the developer guidelines, as it would result in further dead-end entities.

For fun we will try to be able to distinguish police cars in the dataset in the same way that some apps claim to be able to. This is verified with test cases on license plates found online.

4.1.4 Implementation

The concrete coding of the transforms are straight-forward following the resources given in Appendix A and the above design.

We check both API's status messages to determine their availability for continuing execution. `nrpla.de` only sends a status message if something goes wrong and not a HTTP 200 or similar on a successful query; this is handled by a catch-all `try/catch` to look for this and output the error message.

For the domain-transforms we take into account that the registrant and administrator may be either the same or different by a simple `if`-statement on the handle in the beginning. Maltego does help with this, as it group identical entities in the output (i.e. if they have the same main

property).

For the `nrpla.de` transforms, we have had to take measures to avoid both the incompatibility issue with `utf-8` encoded data and the `python 2.x`-methods used and the many fields in the API, which are differing a lot between which vehicle is queried. This is done by checking if fields are `None` and then casting them to `strings`, because that method does not support `None`-types, but the output would otherwise be non-`string` types (`int`'s), which has to be casted, as they do not support the `string.encode("utf8")` edited into the parser to avoid the incompatibility issue.

Apart from this, no other interesting methods have been used when coding, which basically is just parsing from the `json` returned by the API into the appropriate properties of the TRX-framework. A few edits are made based on the test cases, but are better reflected by these (see Sec. 5.1). For additional info on the implementation, please refer to the code (which is commented) or the guidance on Maltego-development in Appendix A.

Neither the documentation from DK Hostmaster or `nrpla.de` was fully up-to-date, so to figure out the exact content, we have used the debug-messages during implementation. These are retained in the code for the interested user to see the data in only a slightly altered format. It is understood that the transforms following the design guidelines should be working and not have the user debug them, but outputting the API's error codes and most of the returned data do give better insight to what may have gone wrong, which it is believed resonates well with the intended audience.

The TRX-framework is essentially a way to build up a temporary mass of data to be parsed as XML and returned to the Maltego client. It offers a few, simple methods that are the only ones we need to use here:

- `ent = TRX.addEntity(String Type, String Value)` To add a new entity-type to the result set.
- `ent.addProperty(String propertyname, String displayname, String matchingrule, String value)` The first argument refers to the property name given on the entity when creating it. The second is only used for *dynamic properties* (when the `propertyname` does not exist on the entity). `matchingrule` allows for entities to be put together, if they are equal (the `strict`-setting is used here – it allows for e.g. similar `thesis.location` to be combined, but nothing else), while `value` is the value of the property.
- `TRX.addUIMessage(String msg, Const type)` is used to return data in the console; it allows for four levels: `debug`, `inform`, `partial` and `fatal`. `fatal` returns a pop-up with the information, while the former are displayed with varying colors (grey, black, orange) in the output console.
- `return TRX.returnOutput()` calls the XML-parser with the concatenated output, which is then returned to the calling transform.

There are more possibilities described in the framework¹², but this is the core functionality. We do not consider a need to display e.g. weights or color the links (except for looks) and the API's are not so elaborate that we e.g. need to read other properties from the input entity.

Both could be necessary in a larger, final set of transforms, which these could be a part of, but not for now, where simple coding can solve the problems.

4.2 The auto-generated report

The auto-generated report is the second delivery of this thesis. It aims to link findings of a OSINT gathering with common attack scenarios and relevant legislation. It does so by providing the security consultant with a framework to input his findings in a standardized format (from the Maltego-client) and label each finding within a range of data categories applicable to OSINT-data. In turn the software generates a report on the findings and its perceived connection with Danish standards and guidelines and a range of common OSINT-enabled cyber attack scenarios (current only targeted attacks).

The link between the data categories, standards/guidelines and scenarios are analysed and designed in Section 3.4.1.

The design of the auto-generated report falls into two parts: Design of the actual report (sections, content, style) is found in Section 4.2.1, while design of the programme is found in Section 4.2.2.

4.2.1 Design of the output report

We discussed requirements for the report in Section 3.4. Based on these requirements, we list the specific elements to implement in this section and their relation with the analysis. Note the similarity to the suggestions of Section 2.1.6 making the report recognizable to its users.

Frontpage The frontpage should contain the name of the author and the client and a “lamp” (inspired by the “lamp” in the Dubex SAC-report and the use of color-grading in both reports (App. B)). It gives the reader an immediate interpretation of the results. This is useful to help him determine the prioritization of reading the report. It is thus important that a red “lamp” is only applied in the worst cases, as the reader otherwise will lose trust of the ratings and may lose confidence in the report's content and its author.

Executive summary contains a few lines about the findings summarizing the amount of scenarios and guidelines the findings are listed against and the disclaimer about which context the results should be considered in (that they are auto-generated and subjective cf. how we necessarily need to make assumptions on the meaning of the data to perform automated linking of findings).

The executive summary is important to easily inform the reader of the result (see Sec. 3.4).

¹²See the documentation on <https://docs.paterva.com/en/developer-portal/reference-guides/trx-library-reference/>

Analogous to the “lamp” on the front page and in the executive summary, it gives a quick interpretation of what is found.

Summary of findings To give an overview of the findings, we summarize them within the data categories later used to be able to uniformly link findings to the scenario and guidelines (see Sec. 3.4.1). We will use a simple histogram showing the distribution between primary categories and frequency tables of findings within each of these categories. This makes the categorization of findings easy to follow and gives a great, easy-to-understand overview.

Summary and findings per scenario Gives a simple explanation of the scenario based on that given in Section 2.5.

It then sums up whether the client are found vulnerable to each scenario in a simple table; it is also here we put the disclaimer on how the report only are able to give a general indication of an increased exposure to an attack scenario.

Next it details the requirements per scenario and the findings linked to each using tables and simple itemized lists. It enables the reader to go deeper into the results, understand how the scenarios apply to the organization (thus raising awareness) and trace the findings back to the origin (as much as the Maltego-export allows – it only exports the value of the parent entity on which the transform was run and the main property of the finding itself).

Summary and findings per standard/guideline In a similar fashion, this section lists the standards/guidelines considered and if they are violated in a simple overview in a table. Next the individual requirements of each standard are listed and the connection with findings are made for easy traceability.

Following the influence the “lamp” may have on the reader’s interpretation of the report, when setting the specific limits it should be considered if the “lamp” are too severe; one could imagine it may convey a sense of negativity or, if the findings are interpreted differently by the reader (whom supposedly have a better insight his own organizational environment), be outright wrong. The length of the auto-generated report should be considered in regards to a reader’s interpretation of the report. Just receiving a long report, can deter a reader. Removing sections of scenarios/standards not found any data on is an approach, but we lose the awareness included by just having them.

For now we decided to implement only the five targeted attack scenarios and picked relevant policies of three standards/guidelines. With three levels, a conservative distinction between the levels can be done such that a violation of at least four scenarios and at least 2 standards *or* all of the scenarios *or* all of the standards results in a “red lamp” and less than either three scenarios and less than 2 standards *or* less than 4 satisfied scenarios/violated standards, results in a “green lamp”.

As the report is *auto-generated* we need to understand the limitations this gives in the design of it. It requires the content to be generated from the findings, which in turn limits possibilities to

create custom content.

The elements of the design as described here all allows for building a general structure in each section, where data can be put in. The executive summary is the biggest challenge in relation to this. Usually one would expect the summary to be custom written based on the findings (see the Dubex SAC-report in App. B.2), but that is difficult with a large domain of results (at least dependent on the sum of all different scenarios and guidelines). To still get the benefit of the auto-generation, we limit the summary to just iterate the findings (e.g. scenarios vulnerable to) and simple sentences, where we can negate the conclusions by inserting e.g. *not*.

Similarly it is not possible to justify conclusions on the client's vulnerability to some scenario further than only listing the findings linked to each requirement. In Section 2.1.6 we noted the usefulness of prioritizing the content to show the most severe cases first, but by using colors and relatively simple and short sections, we find it is not necessary to highlight such. We have neither ranked the scenarios to each other so one can be said to be more severe than the next.

This is acceptable, as the auto-generated report only is an *addition* to the researchers own report. In a later iteration, more text can be written custom to choose from if necessary.

Other extensions include elaborating on the standards in the same way as the scenarios (or even with references) and to explain the labels – supposedly in some kind of appendix.

4.2.2 Designing the program

Designing the program, which is slightly more complex than the Maltego-transforms, we have to consider both the user-aspect as well as designing a reasonable structure that enables for a straightforward flow of execution; being able to understand the flow of execution and the code's purpose, in turn improves maintainability and further development, which is required in this case following Section 3.4.1 concluding that a rework of the data structure behind could be a beneficial next step.

The user intended to generate the report is a security professional with some knowledge of the scenarios, the data labels and what link may exist between them.

The generation should be easy perform, so from the user's actions are as simple as depicted in Figure 4.2. On initiation the program asks the user to select the export file from Maltego, which is then read. Next, he has to label the findings from the input with the labels depicted in the "data categories"-branch in Figure 3.1. We can do that with simple pop-ups, where for each entry in the export, the user has to choose which primary data categories (the first level on the branches of Figure 3.1) fits the finding and next, for each primary category, he chooses none, one or more labels fitting the finding.

After this, the report is generated automatically and a pdf is outputted.

The program has to contain a database of the labels and their associated requirements of both scenarios and standards. A custom data structure is necessary here to allow for easy look-up of the information needed during the report generation.

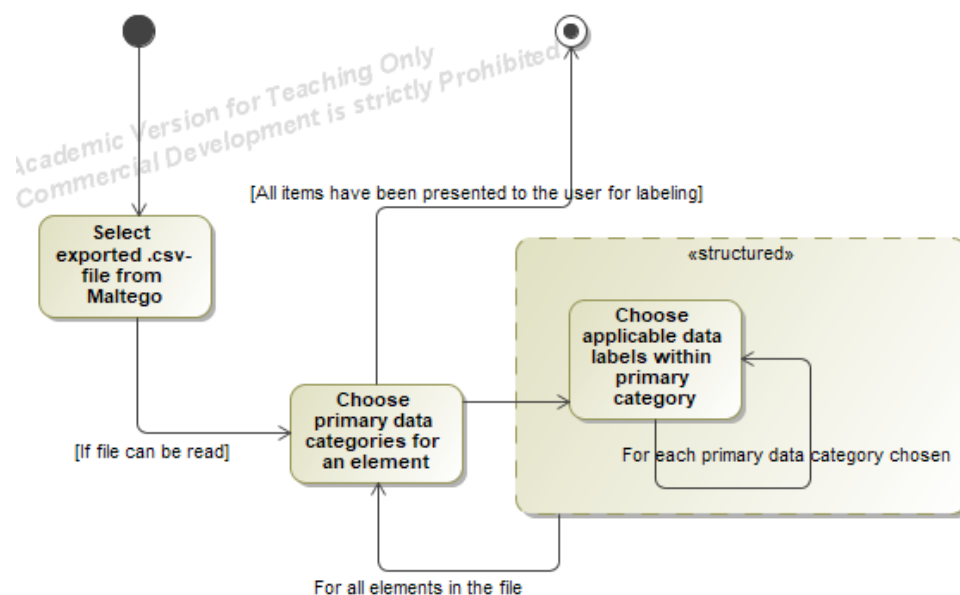


Figure 4.2: The activity diagram for the simple user actions required to generate a pdf-report of findings from Maltego.

In particular we should be able to choose a data label and from that have associated requirements of scenarios and standards returned for generating the conclusions in these. We should also be able to get the labels of each branch to present them to the user during selection of data labels. The objects used should be able to be marked `satisfied` such that we know which scenarios/standards are possible/violated and can list this.

Following the structure of the mindmap allow for this. We design a tree for each primary branch (data categories, scenarios and standards) and let the data categories-branch act as the primary with pointers to leaves of the two other auxiliary trees. This layout is depicted in Figure 4.3.

Each node in the primary tree must have a label, a list of children and leaves and its parent. Each leaf of the primary tree must have a label and references to its parent, a list of scenario requirements and a list of requirements of standards.

The roots in the auxiliary trees must have a list of scenarios/guidelines. These trees always only have one node between the root and the leaves¹³, so each of node (i.e. a standard or scenario) only needs a label, a list of requirements, a count of the number of satisfied requirements and a `boolean` to know if the node itself is satisfied. The two latter saves computation power, as we only have to calculate if some standard/scenario is satisfied once during execution. We can do this, because we only need to perform calculations after the user has read in and labeled all the findings.

For the tree of scenarios, we also need the node to have a static count of how many of its requirements has to be satisfied before we consider the scenario itself possible. This is necessary because several of the news articles and [35] on which the common scenarios are based, not becomes possible just from one finding, but typically requires several OSINT-data to perform.

¹³... for the content to be implemented. Fig. 3.1 rightly shows nested requirements, but they are only sub-labels to distinguish branches within a guideline.

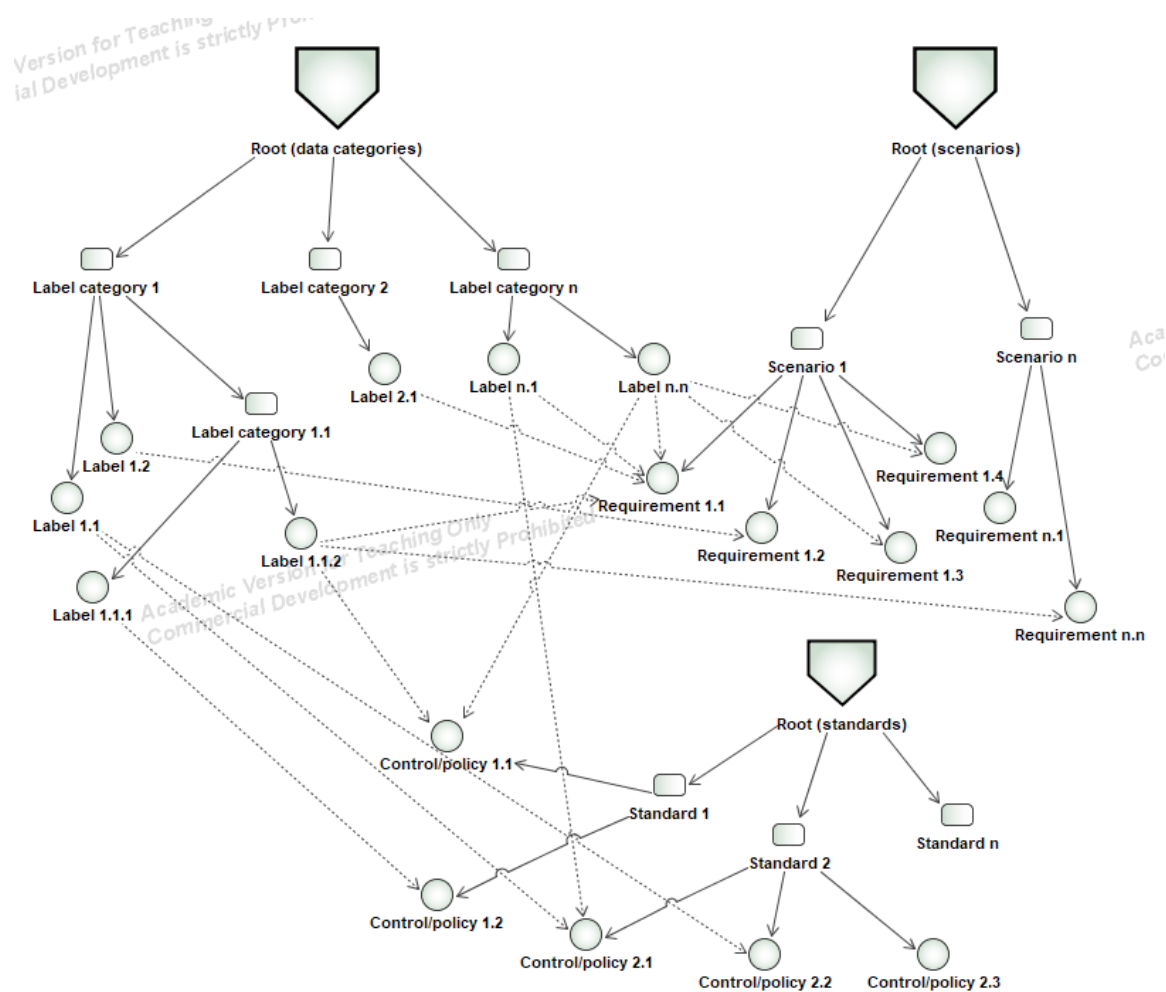


Figure 4.3: The three trees used as the data structure of to hold the data categories, scenarios and standards implemented from the mindmap in Fig. 3.1. The dotted lines indicate pointers from the primary tree of data categories to the two other auxiliaries.

This fact is more pronounced with the slightly more advanced targeted attack scenarios. The leaves in the auxiliary trees only needs a label and a `boolean` to set if they are satisfied following a finding. We discard pointers to the parent scenarios, because it is less computational intensive to just iterate over the scenario’s list of requirements, when we need to calculate if they are satisfied. If we did it the other way around, we would have to walk the entire tree to get the leaves and for each leaf, reference its parent. The top-down approach only needs to iterate all components of the tree once.

An example of how this later was implemented, can be seen in Listing 4.1.

4.2.3 Implementation

As discussed in Section 3.4.1 the difficulty linking data categories, scenarios and standards, prompts that we scope the implementation to the three standards which best relate to actual findings: Mitnick’s guidelines [35], the guidelines of The Federal CIO Council [25] and a small

excerpt of the controls of DS/ISO 27001 [15], which best relates to concrete findings.

Similarly, for the scenarios we only implement common targeted social engineering cyber attacks, which to a larger degree rely on concrete findings and thus can result in more satisfied scenarios, putting some content into the sections.

We will however write the output as if all standards are being considered to demonstrate how the final iteration of the implementation will look.

To enable a cross-platform implementation, we have to consider the choice of code language for the program itself and for generating the report. The code language need to support generating the report.

We need an object-oriented language to make the intended data structure with nodes and leaves; as the transforms are already built for `python`, it is straight-forward to continue with that. It does need an interpreter to run locally, but `python` is widely used¹⁴ so this is acceptable. Due to its widespread use, a lot of packages exists for it. It is for example easy to create the frequency diagram we need using the `numpy`-package.

To create the report, we need to be able to create a `pdf`. Surprisingly no easy frameworks exists for this. To enable cross-platform functionality, built-in libraries for the code language used would be preferred, but they do not exist.

`LATEX` is an alternate, viable choice as it runs on most system architectures and OS's¹⁵.

It is plain-text “code” run through a compiler. The commands are rather simple and it can use input files generated from other parts of the software (we want a frequency diagram among others). Because it is plain-text, we can save `string`-variables of pieces of `LATEX`-code, concatenate them by the necessary logic to output the sections and data listed in Section 4.2.1 and finally parse them to the `LATEX`-compiler.

`python` has a subtype of `strings` for *raw strings*, where backslashes (and thus escape sequences) are not processed (e.g. `\n` resulting in a newline)¹⁶, which is ideal when writing another language within `python` code; thus we can move forward with `python` and `LATEX`.

To create the data structure (step 1-9 of Figure 4.4), we follow the design proposed in Section 4.2.2. An example of how we have implemented this, is found in Listing 4.1.

```
1 # Object for scenarios (nodes)
2 class Scenario(object):
3     name = ""
4     requirements = []
5     required = 0
6     satisfiedRequirements = 0
```

¹⁴4th most used with $\approx 4\%$ share in July 2017 (measured by different search queries for courses, 3rd party vendors etc.) according to <https://www.tiobe.com/tiobe-index/python/> and 2nd most used in practice (measured by tutorials searched on Google) with $\approx 16\%$ share in July 2017 according to <http://pypl.github.io/PYPL.html>.

¹⁵It may even be possible to run it on ARM-architecture? <https://tex.stackexchange.com/questions/115714/latex-for-microsoft-surface>.

¹⁶See https://docs.python.org/3.5/reference/lexical_analysis.html#strings

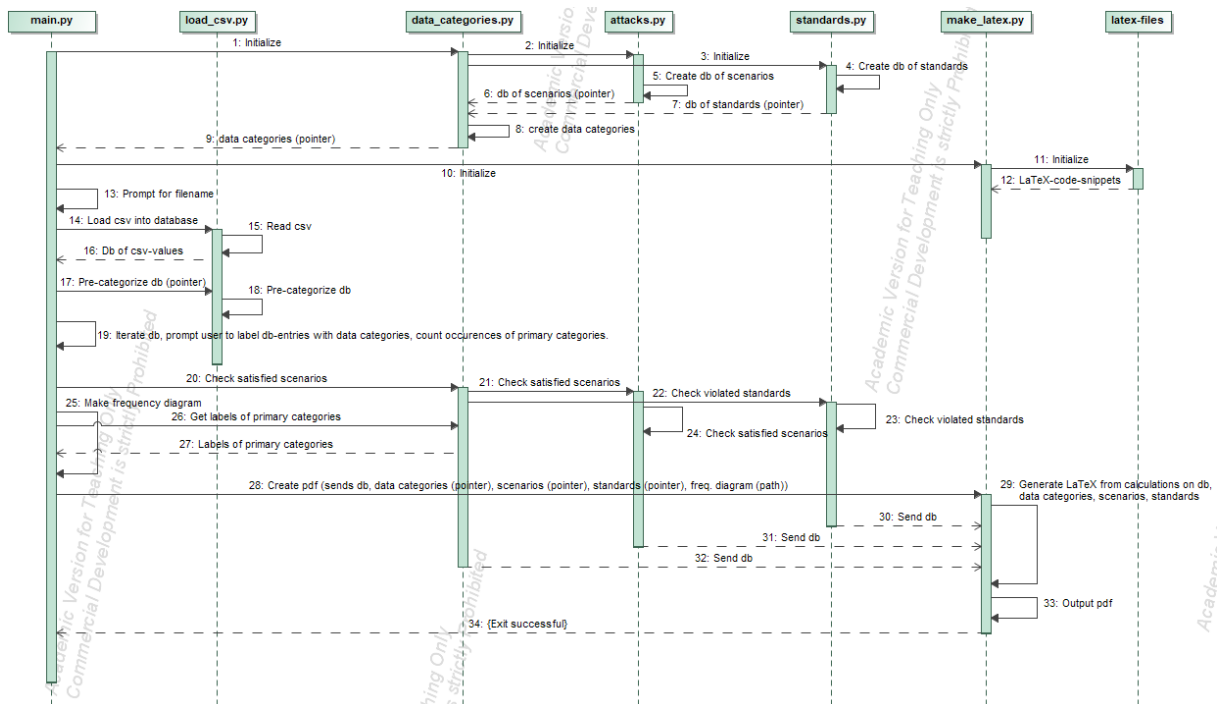


Figure 4.4: A sequence diagram showing interactions between the components of the program for generating the pdf-report.

```

7     satisfied = False
8
9     (...)
10
11     # Compute if scenario is satisfied by checking each of its requirements and
12     # counting if the necessary amount of requirements are satisfied
13     def isSatisfied(self):
14         for requirement in self.requirements:
15             if requirement.satisfied:
16                 self.satisfiedRequirements += 1
17             if self.satisfiedRequirements >= self.required:
18                 self.satisfied = True
19
20     # Object for scenario requirements (leaves)
21     class requirement(object):
22         name = ""
23         satisfied = False
24     (...)
25     ### ROOT ###
26     spearPhishing = Scenario("Spear-phishing", 3)
27     (...)
28     targeted = [spearPhishing, inPerson, ceoFraud, supplyChain, targetedDDoS]
29
30     ##### Targeted attacks #####
31     ### Spear-phishing ###
  
```

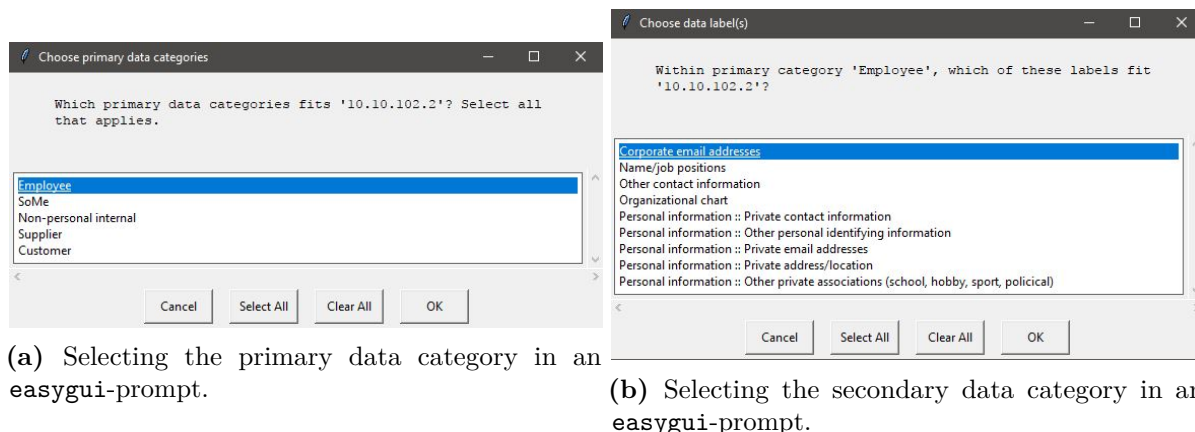


Figure 4.5: easygui-prompts presented to the user for selecting applicable data labels to a finding from the exported csv-file from Maltego. The first option is always pre-chosen.

```
32 sNames = requirement("Employee names/position", spearPhishing)
```

Listing 4.1: The class of scenarios. We see the `isSatisfied()`-method, which updates the state of the data structure by checking if requirements are satisfied. In the bottom, we see how the basic tree structure is created with a node (a scenario) and a leaf (a scenario requirement).

To interact with the user (step 13 and 19 of Fig. 4.4), we want to avoid a lot of weird GUI-coding when we only need simple prompts.

easygui¹⁷ allows for this with simple methods to give choice boxes and other kinds of prompts and simply returning the selected choices/buttons. We e.g. can load the input file with `file = eg.fileopenbox()` and let the user select labels with `choice = eg.multichoicebox(msg, title, choices)`. The first method returns a `string`, the latter a `list of strings` – very simple.

The message containing the finding to be labeled and the choices are generated at run-time by iterating the data structure of labels, first presenting the primary categories and next the labels within the chosen primary categories as seen in Figure 4.5. For labels on deeper levels of the mindmap (thus also in the data structure, which mimicks the mindmap), we prepend their label with the name of the parent node to signify that they are specific labels within that parent-category. An example is the “personal information”-branch under the primary category “employees” as seen on Figure 4.5b.

The first option is always pre-chosen in the prompt and if only one option is sent as an argument to the prompt, it will show a second line “*Add more choices...*”. Selecting it does not affect the returned result though.

To show the data labels we iterate each finding and for each of we iterate all data categories (to get their labels to present in the first prompt) and then get all the leaves of the chosen primary categories (step 19 of Fig. 4.4).

¹⁷<http://easygui.sourceforge.net/>

Running in

$$\mathcal{O}(\text{size}(\text{findings}) * \text{size}(\text{dataLabels}))$$

it is computational heavy, but irrelevant as long as the database structure is so small as here. For more data labels, the choices should be pre-computed.

We develop the program to work with the data/findings that can be expected to be found with the Maltego-platform (or be put in there manually) as discussed by the end of Section 3.4.

We are limited by Maltego's output to the `csv` (see an example in App. D.2). Each line only contains the main property of an entity and its parent; the parent does not have its own line. If no transforms have been run on an entity on the graph, it is included, but only a main property is outputted.

It was considered if it was possible to aid the user even more with the label selection by pre-categorizing (step 18, Fig. 4.4). Due to the lack of information in the export from Maltego we can however only try to guess on the data's type and context of origin. Regular expressions can be utilized to recognize things as URL's, domain names and IP's, which comes in a standardized format, but the context of origin is important to decide between the primary data categories chosen (an example on why it could be beneficial to choose or organize data categories differently as discussed in Sec. 3.4.1). With the current setup, such an approach could only be used to recognize IP's and limit the user to be presented with only the primary data category "non-personal internal" and "suppliers", but even this may restrict the user at some point. Hence we disregarded such a functionality for now.

Next, in step 20 of Figure 4.4 we update the state of the auxiliary data structures of scenarios and standards using the methods reflected in Listing 4.1.

In step 25-27 we generate the frequency diagram.

In step 28, the `LATEX`-generator `make_latex.py` is called with pointers to all three data structures. At program start-up (step 11) `LATEX`-code snippets were read from files containing variables with `LATEX`-code for the different parts of the report, including tables and scenario descriptions; an example is given in Listing 4.2. Notice how the variables containing code to initiate and end the `table`-environment are in separate variables, while a variable contains code for individual lines.

```

1 standards_introTable_start = r"""\begin{{table}}[h]
2 \centering
3 \begin{{tabular}}{|p{{7cm}}|p{{2.5cm}}|}\hline
4 \textbf{{Standard/guideline}} & \textbf{{Violated?}}\\ \hline
5 ""
6 standards_introTable = r"" {label} & \cellcolor{{color}} {satisfied}\\ \hline
7 ""
8 standards_introTable_end = r""\end{{tabular}}\caption{{The standards/guidelines
   considered in this report and whether {client} are considered in violation of
   them.}}
9 \label{{tab:standardOverview}}\end{{table}}
```

Listing 4.2: Example of variables containing L^AT_EX-code to generate a table in a loop in the program generating the report. `r"""..."""` is used to make raw strings over several lines.

Using the `string.format(key = value)`-method we can inject values into the raw string to use for e.g. text or arguments (see Listing 4.3).

The `format`-keys are marked by `{key}`. As L^AT_EX also extensively uses this notation it appears many times throughout the text. As soon the `format`-method is called on a string, all `{...}` are interpreted as keys. To escape `{...}` used for L^AT_EX-commands, we need to use double curly braces: `\command{{argument}}`.

```

1 def makeIt(us, client, freq_diagram, db, infoCategories, attacks, standards):
2     [a,s,text,lamp] = calculateSatisfiedScenarios(attacks,standards)
3     latex = r""" """
4     latex += preamble_frontpage.format(client = client, us = us, lamp = lamp,
5     noScenarios = len(attacks), noStandards = len(standards), satisfiedScenarios =
6     a, satisfiedStandards = s, good = text, noFindings = len(db))
7     latex += intro.format(client = client, us = us, noCategories = len(
8     infoCategories), noScenarios = len(attacks), noStandards = len(standards))
9     latex += stats_intro.format(freqdiagram = freq_diagram, noCategories = len(
10    infoCategories))
11    latex += generateStats(infoCategories, db)
12    latex += generateScenarios(infoCategories, db, attacks, client)
13    latex += generateStandards(infoCategories, db, standards, client)
14    latex += endDoc
15    (...)

```

Listing 4.3: Example of how the variables containing L^AT_EX-code is formatted and concatenated into one variable containing all code

The program also injects simple statements into the text such as small conclusions (“satisfactory, but room for improvement”) or negations (“not”) to re-use as much L^AT_EX-code as possible and avoid a lot of individual code only used under specific conditions.

`make_latex.py` loops over the scenarios and standards respectively to generate the necessary L^AT_EX-code from the categorized findings. This is a bit computationally heavy, especially in the section detailing the data that were found to be linked with some requirement of the scenario/standard. This requires to iterate all requirements of all scenarios/standards (for each section we are going to fill in linked findings for), check all labels of all findings, to get all leaves of each primary data category, if the label of a leaf is used on the finding, we iterate its linked requirements and if this matches the current section we are filling in findings of, we can input the finding (and its parent, if any).

It is a backwards approach and the only place where the data structure inhibits an effective algorithm. Alternatively the program should compute this at run-time, but for were the other approach uses more computational resources, this approach will use more space. An improved

data structure is necessary to support this operation; the current entails a choice between which of the two resources we would rather put to use.

The \LaTeX -code is generated sequentially section-by-section using input from the `.py`-files containing variables with \LaTeX -code in `raw string`. We see how this approach together with re-use of the \LaTeX -code, limits elaborate conclusions and summaries in human words taking into perspective many different aspects that may have arised during the investigation. The goal of the auto-generated report was however not to make report able to be used as a stand-alone delivery by consultants, but just as a part of the investigation's deliveries to the customer; these overall conclusions will come from the security researcher himself and ours is an input hereto. There are some very weird behavior using `raw strings`. Large blocks of text is necessary to make the custom explanations for the scenarios. While we can put \LaTeX -code into separate variables in `raw strings`, this is not possible to read from e.g. a file. The latter approach would enable a simple to contain the explanation for each scenario and give the path as an argument to the `scenario`-object, which in turn could be read and added to the \LaTeX -code during concatenation.

For no reason this approach does not work and what the reason is, is not clear. Instead we hard-code an array of `raw strings` in the `.py`-file for variables to make the scenario-section. It uses the same positions as the list of each `scenario`-object enabling us to pick it consecutively from there at run-time.

Having concatenated the \LaTeX -code, it is parsed as an argument to `pdflatex` for compilation. We remove the auxiliary \LaTeX -files using `os.unlink("file")`. Both procedures are shown in Listing 4.4.

```
1 def makeIt(us, client, freq_diagram, db, infoCategories, attacks, standards):
2     (...)
3     # Write LaTeX-code to file
4     with open('output.tex', 'w') as f:
5         f.write(latex)
6
7     for i in range(0,2): # Run pdflatex twice to update references
8         cmd = ["pdflatex", '-interaction', 'nonstopmode', 'output.tex']
9         proc = subprocess.Popen(cmd)
10        proc.communicate()
11
12        retcode = proc.returncode
13        if not retcode == 0: # Error handling; removes the .pdf if this happens,
14        as it may be incorrect.
15            os.unlink('output.pdf')
16            raise ValueError('Error {} executing command: {}'.format(retcode, ' '.
17        join(cmd)))
18
19    # Delete auxiliary files
```

```
18 os.unlink('output.tex')
19 os.unlink('output.log')
20 os.unlink('output.toc')
21 os.unlink('output.out')
22 os.unlink('output.aux')
```

Listing 4.4: Example of how the variables containing \LaTeX -code is parsed to the \LaTeX -compiler.

A example of the final report is found in Appendix B.3.

Chapter 5

Tests

This chapter describes the tests that are necessary to perform to ensure that the developed products adhere to requirements set up in the analysis (Chapter 3). This includes ensuring reliable execution and that the products produce the expected output.

5.1 Transforms

The transforms cannot be run directly in the IDE, thus neither tested there as one would do with e.g. unit tests. Instead we need to call the transforms from within Maltego on a variety of entities and compare the actual and expected output. We do this with a “grey-box” approach, i.e. we know the inner workings of our code and what types of results can be expected from the API’s, but parts in-between are closed to us (e.g. the parser and Maltego’s treatment of the resultset).

The aim of the test is to ensure that the transforms follow the requirements of Chapter 3: They need to return the expected entities containing the expected information in the correct property-fields; if this fails, we need to ensure that proper error messages are displayed to the user. This is also in line with the design guidelines [44] (part of the requirements).

We will also need to test that all the desired entities listed in Section 4.1.2 and 4.1.3 are returned correctly. This is a core requirement for the transforms to be “useful” and add value to an OSINT-investigation.

The test cases for the two different providers are outlined in two sections below, as they are not entirely identical due to difference in the resultset including data types and encoding used. We list which input has been used for the test, but as the result is purely visual in Maltego and the results appear in part both on the graph, in the properties of each entity and in the console, we primarily note if the tests have been passed by the transform and include figures illustrating the result for all tests of their output relevant to the test.

As noted earlier, we expect the API-providers to adhere to their documentation; that is, the keys and values returned are as described and will remain so for at least the duration of this project. For the API of `nrpla.de` we however observed a lack of documentation for several of the specialty

fields in the result set. The documentation only has one example per URL/registry and not all fields are used in this example. This prompts us to test all categories of vehicles to get an output in these fields and ensure proper implementation.

During the development we have identified (manually and from the documentation) two major problem with the XML-parser provided by Paterva (see outline of how this runs in Fig. 4.1): It is coded to work with `python 2.x` and thus differently from most other implementations by now¹ and it lacks proper escaping of special characters, that has to be done in XML-code.

Both problems have required long email-conversations with Paterva to solve the issues as well as debugging their parser. In particular this resulted in problems with handling non-ascii characters and values set to `None` (`null` in python). Both problems are reflected in the tests and are now solved.

We also identified problems with the implementation of a new feature in the entity properties in Maltego. Paterva have made a default values in properties of some entities (e.g. `maltego.Location`), which uses values of other properties. These however overrides setting the property manually such that it is no longer possible to use the `name`-property (the main property of this particular entity). Instead we have to add a separate property and assign this as a new main property to display a value, which better conveys the information in the entity. An example can be seen in Figure 5.1, where the property `Display name` is used for the actual name, while the property `Name` is a concatenation of the properties `City` and `Country`, which we do not want here.

According to Paterva, while it is not the best solution, it is expected behavior to overrule the developers intentions.

Tests for basic cases are implied from all the irregular cases contained in the tests below, e.g. for DK Hostmaster we test that data from fields in the result which are usually `None` are displayed correctly in Maltego – the cases for when they *are None* are implied by the other tests, as this is the norm; in these cases no data will be returned and we only need to know, that transform runs successfully.

The test cases also overlap in some of the cases, but this only ensures further robustness of the code.

5.1.1 DK Hostmaster-transforms

To test the implementation of the domain-transform based on DK Hostmaster's API, we run the transform on different domains as listed in Table 5.1. We look for input to test availability of all entites listed in Section 4.1.2 and the errors found in the platform as noted in Section 5.1.

For the handle-transforms, we test the cases as listed in Table 5.2.

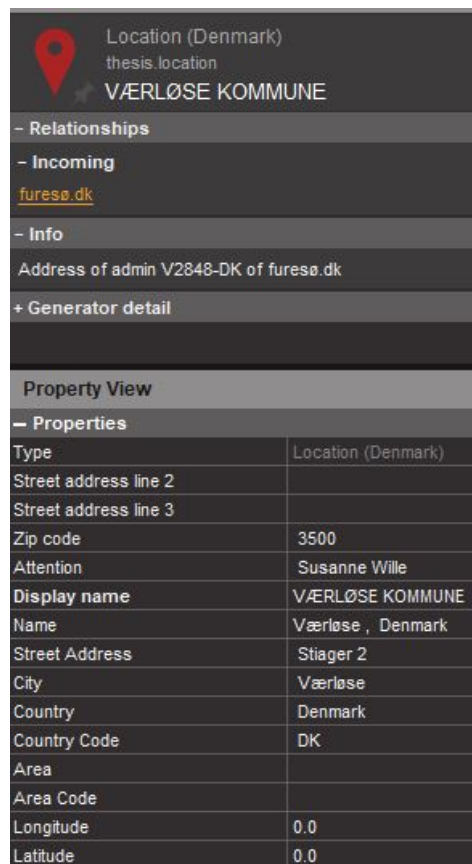
¹The transition has been slow, but already some three years ago, `python 3.x` was the appropriate choice for future-proofing code.

Test	Explanation	Input	Passed?
1: An input with non-ascii characters.	All characters used by DK Hostmaster should be supported; our code expects <code>utf-8</code> , which DK Hostmaster supports, but Paterva's parser do not out-of-the-box. Input with non- <code>ascii</code> also results in testing output for correct handling.	<code>maltego.Domain</code> with value <code>furesø.dk</code> (contains <code>æ</code> , <code>ø</code> , <code>å</code>). We assert the rest of the character set is working too.	OK (see Fig. 5.1)
2: An invalid domain.	Run the transform on a non- <code>.dk</code> or non-existent domain; we do not check if is valid, but DK Hostmaster will not return <code>HTTP 200</code> if the domain was invalid or non-existent.	<code>maltego.Domain</code> with value <code>garbage</code> .	OK (see Fig. 5.2)
3: A domain, where the handle uses an uncommon field.	Run a transform on a <code>.dk</code> -domain, where the handle uses a field that is <code>None</code> in most cases; here the <code>attention-field</code> .	Covered by test 1, Table 5.1.	OK
4: A domain, where the handle of the registrant and admin is protected	If the person behind the handle has protected address in CPR, it does not show up on DK Hostmaster, but still contains some data.	<code>maltego.Domain</code> with value <code>ninabrorson.dk</code>	OK (see Fig. 5.3)
5: A personal domain	A domain registered to an individual to check if entities on a person is generated.	<code>maltego.Domain</code> with value <code>mtborientering.dk</code> .	OK (see Fig. 5.4)
6: A domain not belonging to an individual	A domain not registered to an individual to check that fields are populated correctly with this information.	Covered by test 1, Table 5.1.	OK
7: A domain with a telephone number on the account	Only very few has this. The implementation of each of them are identical, so if one works, the rest are asserted to do as well.	Covered by test 5, Table 5.1.	OK

Table 5.1: Test-cases for the DK Hostmaster-domain transforms.

Especially test 4, Table 5.1 and test 6, Table 5.2 demonstrates how this method of testing are difficult to ensure complete coverage. I was lucky to identify an account and a domain with this property and it also relied on prior knowledge of the domain registry.

Despite this, the above test covers most (if not all) possible input-types to the transform, so they ensure adequately reliable transforms for proper usage. We have not put “regular” cases separately in the test overview, but their function follows from the successful tests performed, e.g. test 3, 5 and 6 in Table 5.1 do test “special” cases, but the three of them covers virtually all input types.



Location (Denmark)
thesis.location
VÆRLØSE KOMMUNE

- Relationships

- Incoming
[furesø.dk](#)

- Info
Address of admin V2848-DK of furesø.dk

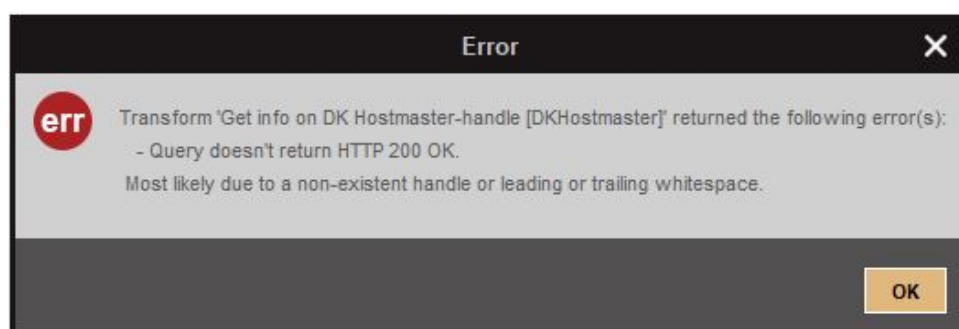
+ Generator detail

Property View

- Properties

Type	Location (Denmark)
Street address line 2	
Street address line 3	
Zip code	3500
Attention	Susanne Wille
Display name	VÆRLØSE KOMMUNE
Name	Værløse , Denmark
Street Address	Stiager 2
City	Værløse
Country	Denmark
Country Code	DK
Area	
Area Code	
Longitude	0.0
Latitude	0.0

Figure 5.1: In-/output of domain with non-ascii characters.



```
Running transform Get info on DK Hostmaster-handle [DKHostmaster] on 1 entities (from entity "garbage")
Read handle: garbage and type thesis.dkhostmaster.handle (from entity "garbage")
key: status    value: 400 (from entity "garbage")
key: message   value: Bad Request (from entity "garbage")
```

Figure 5.2: Errors returned running test 2 of Table 5.1. The error gives a possible solution and the output lists the input (if debug-messages are enabled by the user).

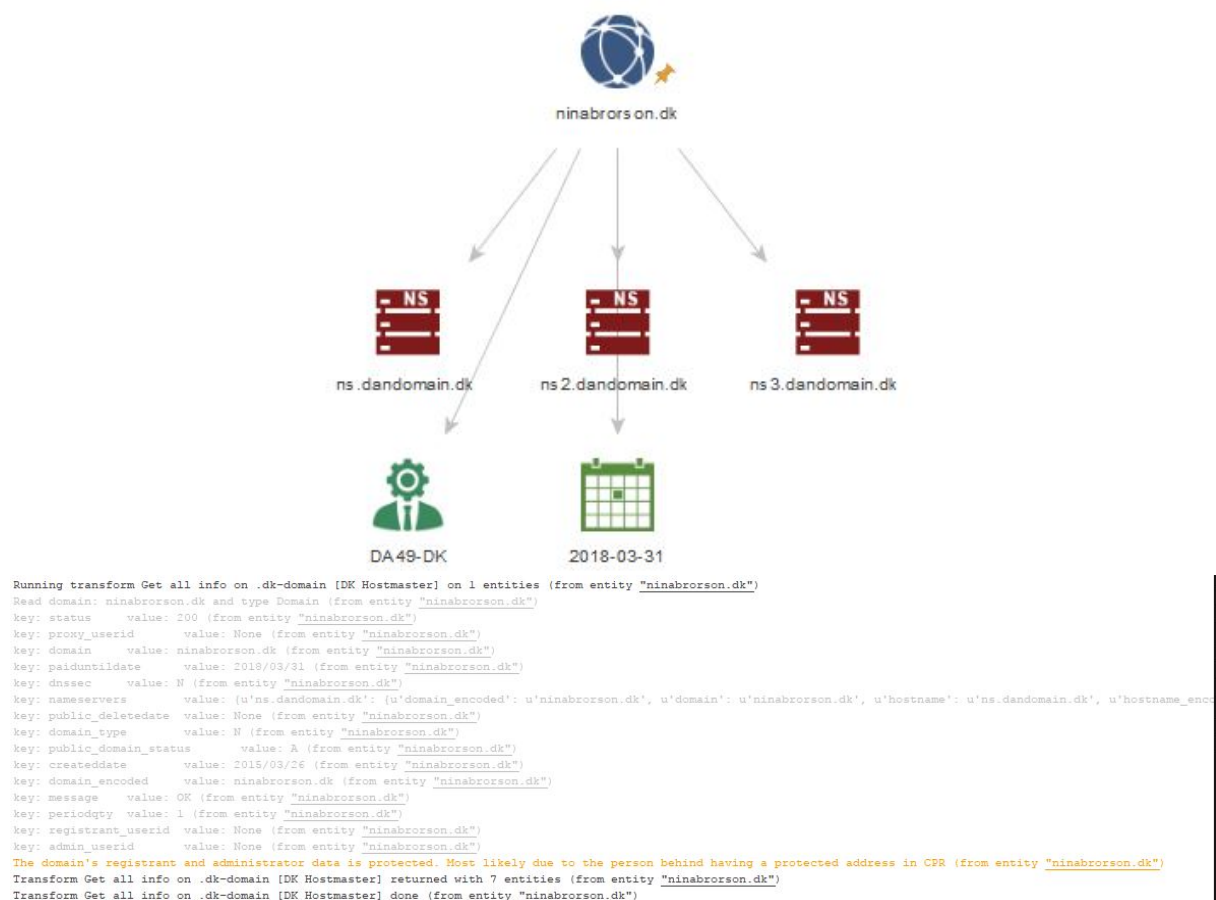


Figure 5.3: Results of running test 4 of Table 5.1. We return a *partial error* telling of the finding, but return the rest of the information found. The debug-output lists the findings.

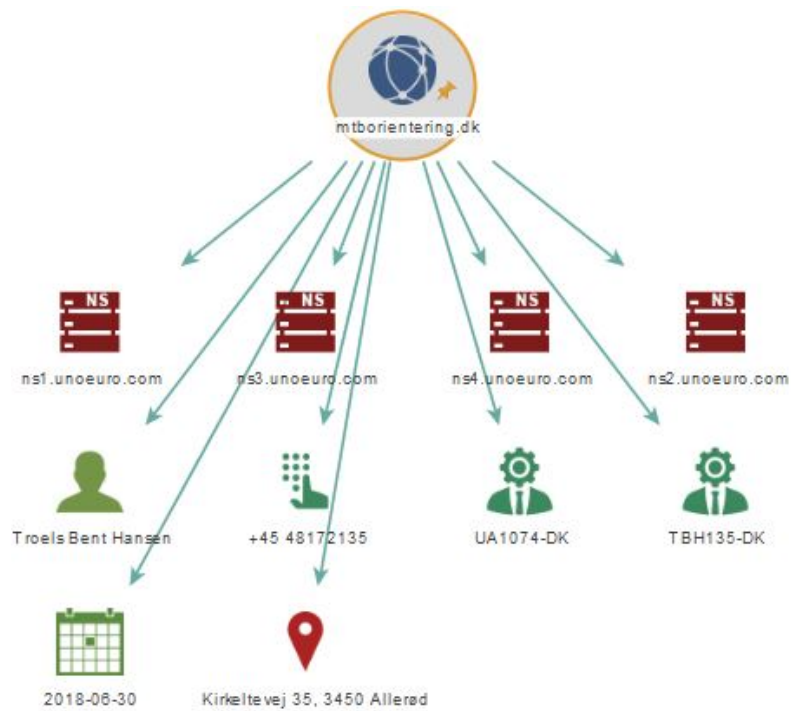


Figure 5.4: A domain with a *Personal* account. Besides the same entities as in other tests, we note the entity for a person, a location for him and a telephone number.

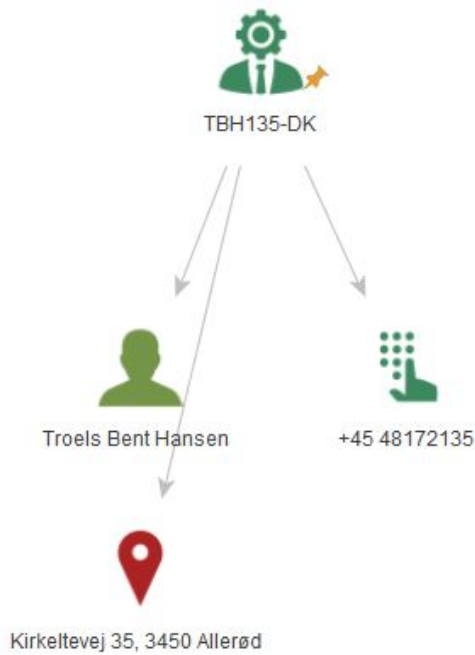
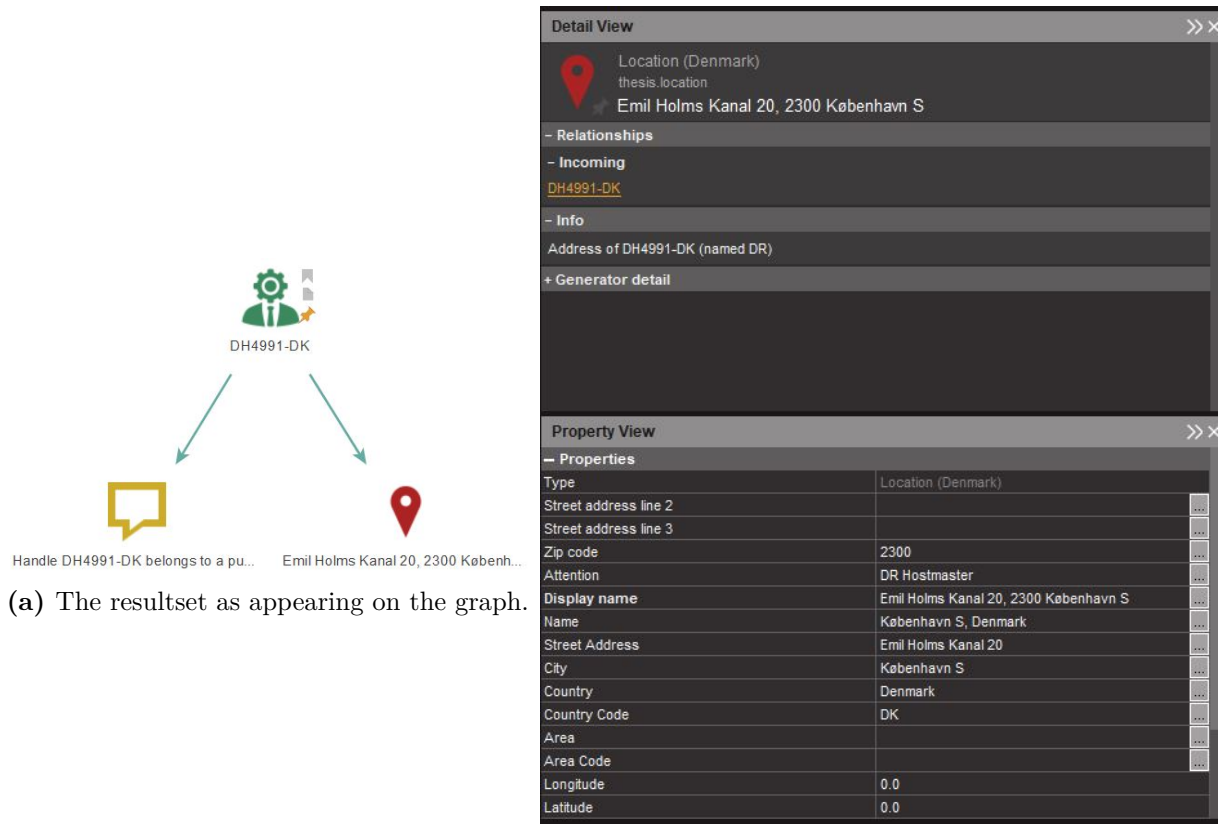


Figure 5.5: The resultset on the graph from running test 1, Table 5.2.

Test	Explanation	Input	Passed?
1: An existing handle registered to a person.	Tests the use of the <code>Person</code> -entity.	<code>thesis.dkhostmaster.handle</code> with value <code>TBH135-DK</code> .	OK (see Fig 5.5)
2: An existing handle not registered to a person.	Tests the use of the <code>Phrase</code> -entity, which is used when the account is non- <i>Personal</i> .	<code>thesis.dkhostmaster.handle</code> with value <code>DH4991-DK</code> (this is a public sector organization).	OK (see Fig. 5.6)
3: An existing handle using an uncommon field.	Tests implementation of fields in the result set, which are usually <code>None</code> .	Included in test 2, Table 5.2.	OK
4: A non-existing handle.	As in Table 5.1 this aims to test that errors from DK Hostmaster are handled appropriately.	<code>thesis.dkhostmaster.handle</code> with value <code>garbage</code>	OK (see Fig. 5.7)
5: A handle using non-ascii characters.	The same test as in Table 5.1.	Included in test 1, Table 5.2.	OK
6: A protected handle	If the person behind the handle as protected address in CPR, it does not show up on DK Hostmaster.	<code>thesis.dkhostmaster.handle</code> with value <code>NB5655-DK</code>	OK (see Fig. 5.8)
7: A handle with telephone number	Tests the implementation of telephone numbers similarly to test 7, Table 5.1	Covered by test 1, Table 5.2	OK

Table 5.2: Test-cases for the DK Hostmaster-handle transforms.



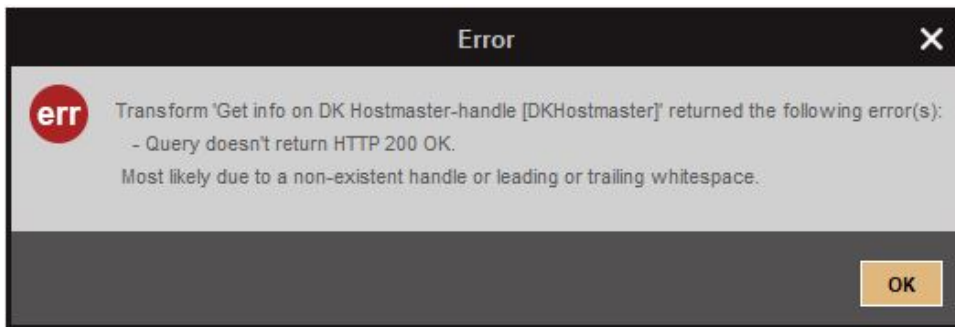
(a) The resultset as appearing on the graph.

(b) The detailed content of the thesis.location-entity.

```
Running transform Get info on DK Hostmaster-handle [DKHostmaster] on 1 entities (from entity "DH4991-DK")
Read handle: DH4991-DK and type thesis.dkhostmaster.handle (from entity "DH4991-DK")
key: status      value: 200 (from entity "DH4991-DK")
key: city        value: København S (from entity "DH4991-DK")
key: street3     value: None (from entity "DH4991-DK")
key: mobilephone value: None (from entity "DH4991-DK")
key: validregistrant value: 1 (from entity "DH4991-DK")
key: telefax     value: None (from entity "DH4991-DK")
key: street1     value: Emil Holms Kanal 20 (from entity "DH4991-DK")
key: street2     value: None (from entity "DH4991-DK")
key: attention   value: DR Hostmaster (from entity "DH4991-DK")
key: userid      value: DH4991-DK (from entity "DH4991-DK")
key: query_userid value: DH4991-DK (from entity "DH4991-DK")
key: zipcode     value: 2300 (from entity "DH4991-DK")
key: phone       value: None (from entity "DH4991-DK")
key: countryregionid value: DK (from entity "DH4991-DK")
key: useridtype  value: 0 (from entity "DH4991-DK")
key: message     value: OK (from entity "DH4991-DK")
key: name        value: DR (from entity "DH4991-DK")
Transform Get info on DK Hostmaster-handle [DKHostmaster] returned with 2 entities (from entity "DH4991-DK")
Transform Get info on DK Hostmaster-handle [DKHostmaster] done (from entity "DH4991-DK")
```

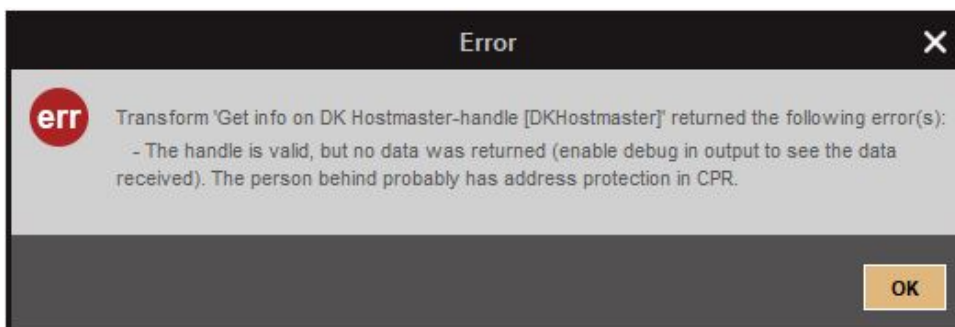
(c) The console output to the Maltego-client (light-gray is the debug messages).

Figure 5.6: Results from running test 2 of Table 5.2. We see a phrase-entity is returned, that the location-entity is filled out correctly and can see the original resultset from DK Hostmaster (if debug-messages are enabled by the user).



```
Running transform Get info on DK Hostmaster-handle [DKHostmaster] on 1 entities (from entity "garbage")
Read handle: garbage and type thesis.dkhostmaster.handle (from entity "garbage")
key: status      value: 400 (from entity "garbage")
key: message     value: Bad Request (from entity "garbage")
```

Figure 5.7: Errors returned running test 4 of Table 5.2. The error gives a possible solution and the output lists the input (if debug-messages are enabled by the user).



```
Running transform Get info on DK Hostmaster-handle [DKHostmaster] on 1 entities (from entity "NB5655-DK")
Read handle: NB5655-DK and type thesis.dkhostmaster.handle (from entity "NB5655-DK")
key: status      value: 200 (from entity "NB5655-DK")
key: city        value: None (from entity "NB5655-DK")
key: street3     value: None (from entity "NB5655-DK")
key: mobilephone value: None (from entity "NB5655-DK")
key: validregistrant value: 1 (from entity "NB5655-DK")
key: telefax     value: None (from entity "NB5655-DK")
key: street1     value: None (from entity "NB5655-DK")
key: street2     value: None (from entity "NB5655-DK")
key: attention   value: None (from entity "NB5655-DK")
key: userid      value: None (from entity "NB5655-DK")
key: query_userid value: NB5655-DK (from entity "NB5655-DK")
key: zipcode     value: None (from entity "NB5655-DK")
key: phone       value: None (from entity "NB5655-DK")
key: countryregionid value: None (from entity "NB5655-DK")
key: useridtype value: None (from entity "NB5655-DK")
key: message     value: OK (from entity "NB5655-DK")
key: name        value: None (from entity "NB5655-DK")
Transform Get info on DK Hostmaster-handle [DKHostmaster] returned with 0 entities (from entity "NB5655-DK")
Transform Get info on DK Hostmaster-handle [DKHostmaster] done (from entity "NB5655-DK")
```

Figure 5.8: Errors returned running test 6 of Table 5.2. The error gives a plausible explanation and the output lists the input (if debug-messages are enabled by the user). Note how it returns 200, but None in all fields containing information about the handle.

5.1.2 License plate transform

To test the transforms for `nrpla.de`, we exploit a bit of knowledge of the Danish registration system. The data on each vehicle varies a lot between type and especially between new and older passenger vehicles after some changes to the system in the end of the 00's. The approach is thus to run the transform on all kinds of vehicles that are issued plates for in Denmark. We have to exclude special issues like vehicles of the royals or the emergency services, as their license plates cannot be looked up² and we have not been able to find VIN's of these.

This approach is bit different from the DK Hostmaster-transforms, since there e.g. a difference in account types only meant a chance in one field, while it here cannot be known if some other field is set in a proprietary way.

There are a lot of fields to be used in the data. Appendix D.1 shows an example of these – specifically the output of test 1, Table 5.3.

Test cases for every single field is not made, but with the focus on the different vehicle categories, we expect adequate coverage. The issue remains as with the DK Hostmaster API, that some obscure case might exist, but lessons learned from the other test cases are implemented appropriately and will catch other cases as well. An example: Some field is `None` and fails returned to the parser without being cast to a `string`. Recognizing this, we harden all the code reading the data, where the fields can be `None` to handle this.

To verify the transform for most of other types vehicles, we have used the input as show in Table 5.3. Especially with the specialty types (trailers, scooters, government vehicles) we need to ensure that the fields display properly and that we can output the information found. References to the screenshots of the results are given for all test cases shown on the following pages.

We have not included the debug information in the console, as `nrpla.de` does not include elaborate error messages than cannot be contained in a pop-up. The output is identical to that shown in Appendix D.1.

Test	Explanation	Input	Passed?
1: A passenger vehicle first registered before extended information.	A specific date is hard to find, but it happened between 2008 and 2012.	<code>thesis.licenseplate</code> with value <code>BB29177</code> (registered in 2005)	OK (see Fig. 5.10)
2: A passenger vehicle registered with extended information.	We have not implemented all fields (see field implementation scheme in App. D.1), but there are more info to be found for newer cars.	<code>thesis.licenseplate</code> with value <code>AD25729</code> (registered in 2013)	OK (see Fig. 5.11a)

²Examples are the DEMA (“Danish Emergency Management Agency” or “Beredskabsstyrelsen”) using plates with 5 numbers, fire brigades either have regular plates or something like `M1`, the royals use e.g. `Krone 1`

3: A leased passenger vehicle	Leasing has a separate field.	<code>thesis.licenseplate</code> with value BP22043 (leased for 3 years from now)	OK (see Fig. 5.11b)
4: A vehicle with debt	Tests the amount, the creditor and debtors are correctly displayed. Most cars are without publicly notated debt, so the opposite case is covered.	<code>thesis.licenseplate</code> with value AB12345 has debt to DONG Energy A/S	OK (see Fig. 5.12)
5: A vehicle with no inspections	In the API this cause the field to be missing. Inspections only lack for brand new cars.	Covered by test 3, Table 5.3	OK
6: A vehicle where the license plates are revoked	Creates issues especially for getting the tax on the car, as no present amount is shown and at least the first amount listed historically will be a refund.	<code>thesis.licenseplate</code> with value AM51978	OK (see Fig. 5.13)
7: A motorcycle		<code>thesis.licenseplate</code> with value ET11900	OK (see Fig. 5.14a)
8: A scooter		<code>thesis.licenseplate</code> with value OJ109	OK (see Fig. 5.14b)
9: A trailer		<code>thesis.licenseplate</code> with value AH9433	OK (see Fig. 5.15a)
10: A caravan		<code>thesis.licenseplate</code> with value YJ2955	OK (see Fig. 5.15b)
11: A police car	Apps exist that claim to be able to distinguish them due to a special registration.	<code>thesis.licenseplate</code> with value AL45776	OK (see Fig. 5.16a)
12: An ambulance	To see another kind of government vehicle and uncover any special implementations of fields.	<code>thesis.licenseplate</code> with value AV89845	OK (see Fig. 5.16b)
13: A lorry		<code>thesis.licenseplate</code> with value TY88978	OK (see Fig. 5.17a)
14: A bus		<code>thesis.licenseplate</code> with value BG88467	OK (see Fig. 5.17b)

15: A vehicle with several different inspection locations	To verify that the different locations are displayed properly and convey the historic info appropriately	Covered by test 1, Table 5.3	OK (included in Fig. 5.10)
16: A VIN-number	All the cases are independent of whether a VIN or license plates are used as input, but we have to ensure a VIN can be used.	<code>thesis.vin</code> with value <code>WOLBF8EC2G8067766</code> , which also hit a rare case where no creditor is listed even though it has debt. We were not aware of such a construct.	OK (see Fig. 5.18)
17: An invalid license plate	The error handling of invalid license plates and VIN's are identical, so we only need to test one of them	<code>thesis.licenseplate</code> with value <code>Krone2</code>	OK (see Fig. 5.19)
18: A license plate with letters	Denmark allows for custom license plates between 2 and 7 characters (numbers and letters).	<code>thesis.licenseplate</code> with value <code>ESTES</code>	OK (see Fig. 5.9)

Table 5.3: Test-cases for the transform for Danish license plates/VIN's on the API of `nrdpla.de`.

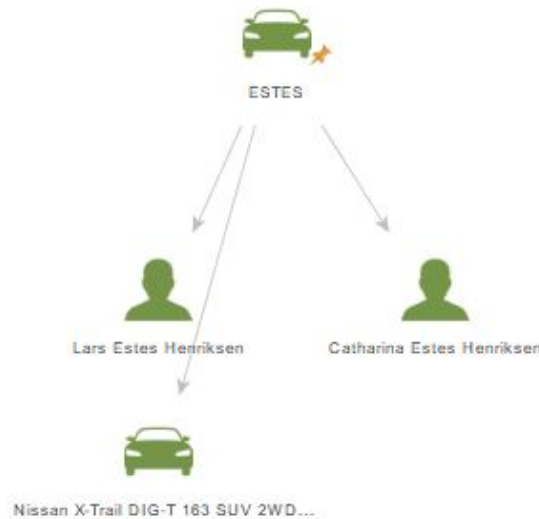


Figure 5.9: Graph view of a car with a custom license plate (and debtors).

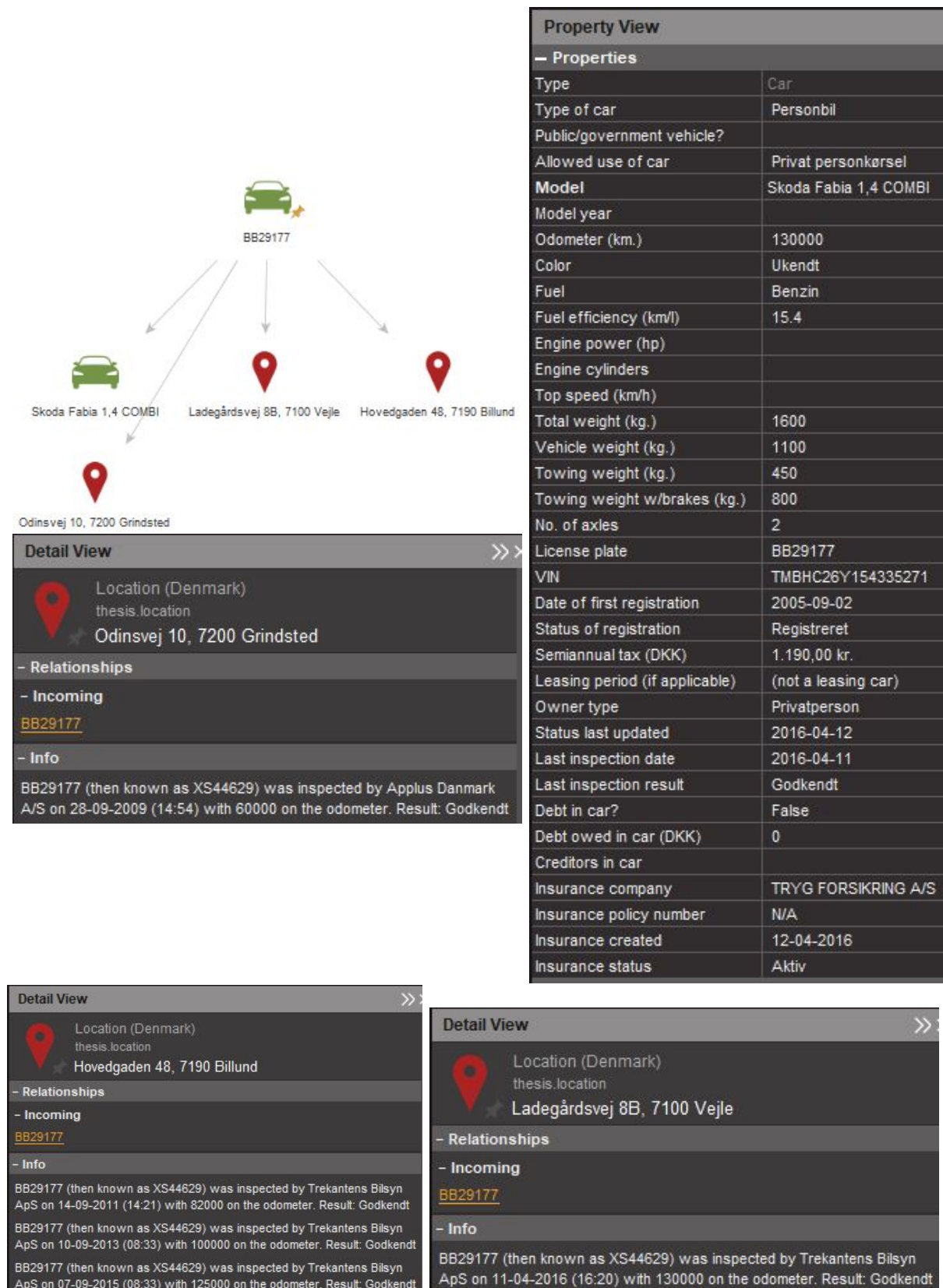


Figure 5.10: The output on the graph from a car before the extended fields were used in the registry; note how a lot of properties are empty. On the graph we also see the location-entities of the inspection halls. Three screenshots shows the property view of each location-entity for the inspection halls, demonstrating the ability to tie the vehicle to a specific location (at some point of time) due to several inspections in the same are (cf. test 15, Table 5.3). Historic license plates of the vehicle are included in this data.

Property View		Property View	
— Properties		— Properties	
Type	Car	Type	Car
Type of car	Personbil	Type of car	Personbil
Government vehicle?		Government vehicle?	
Allowed use of car	Privat personkørsel	Allowed use of car	Privat personkørsel
Model	Ford B-Max 1.0 EcoBoost (100 HK) Hatchback	Model	Kia Ceed 1.6 CRDI Stationcar Man. 6
Model year	2012	Model year	2017
Odometer (km.)	17000	Odometer (km.)	0
Color	Grå	Color	Sort
Fuel	Benzin	Fuel	Diesel
Fuel efficiency (km/l)	20.4	Fuel efficiency (km/l)	25.6
Engine power (hp)	74	Engine power (hp)	100
Engine cylinders	3	Engine cylinders	4
Top speed (km/h)	175	Top speed (km/h)	194
Total weight (kg.)	1760	Total weight (kg.)	1920
Vehicle weight (kg.)	0	Vehicle weight (kg.)	0
Towing weight (kg.)	635	Towing weight (kg.)	650
Towing weight w/brakes (kg.)	750	Towing weight w/brakes (kg.)	1500
No. of axles	2	No. of axles	2
License plate	AD25729	License plate	BP22043
VIN	WF0KXXERJKCY16724	VIN	U5YHN816AHL229758
Date of first registration	2013-02-11	Date of first registration	2017-07-19
Status of registration	Registreret	Status of registration	Registreret
Semiannual tax (DKK)	310,00 kr.	Semiannual tax (DKK)	1.060,00 kr.
Leasing period (if applicable)	(not a leasing car)	Leasing period (if applicable)	2017-07-19 to 2020-08-18
Owner type	Privatperson	Owner type	Privatperson
Status last updated	2013-02-11	Status last updated	2017-07-19
Last inspection date	2017-02-07	Last inspection date	None
Last inspection result	Godkendt	Last inspection result	
Debt in car?	False	Debt in car?	False
Debt owed in car (DKK)	0	Debt owed in car (DKK)	0
Creditors in car		Creditors in car	
Insurance company	Alm. Brand	Insurance company	GJENSIDIGE FORSIKRING
Insurance policy number	N/A	Insurance policy number	N/A
Insurance created	26-01-2016	Insurance created	19-07-2017
Insurance status	Aktiv	Insurance status	Aktiv

(a) The property view of a newer car using the extended fields. Notice how all properties regarding car specifications are filled out.

(b) The property view of a leased car. Notice the **leasing period**-property correctly displaying the duration of the lease contract and no **Last inspection date** is returned (because its a brand new car).

Figure 5.11: The property view of a new and a leased car side-by-side.

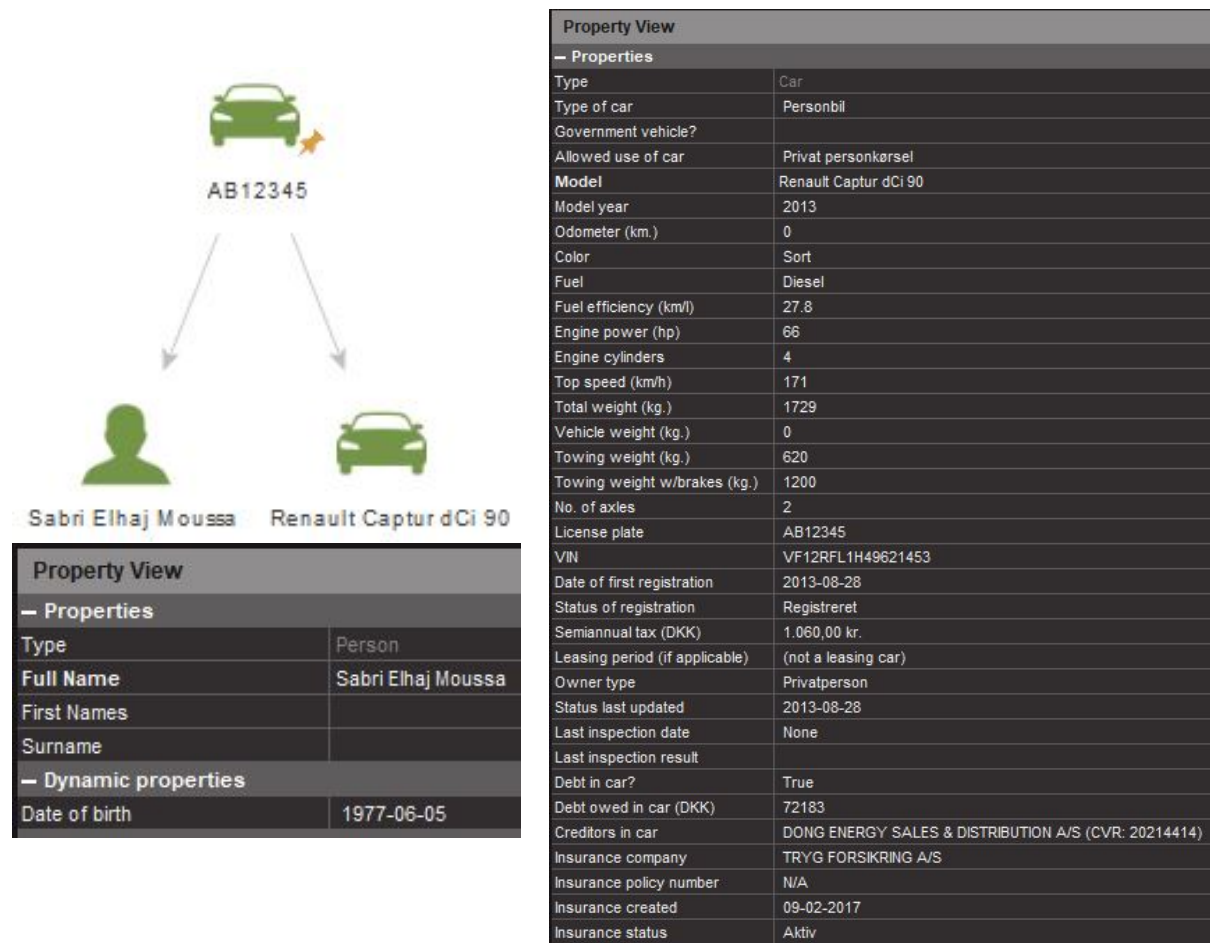


Figure 5.12: A car with debt. Notice a `maltego.Person` entity is returned with the debtor and we have included the debtor's date of birth in the property view. In the property view of the car, the name and CVR of the creditor is included too (and can later be extended into its own entity, when the CVR-transforms gets implemented).

Property View	
— Properties	
Type	Car
Type of car	Personbil
Government vehicle?	
Allowed use of car	Privat personkørsel
Model	Skoda Citigo 1.0 MPI 60 HK 5-Dørs Hatchback
Model year	2014
Odometer (km.)	0
Color	Hvid
Fuel	Benzin
Fuel efficiency (km/l)	24.4
Engine power (hp)	44
Engine cylinders	3
Top speed (km/h)	161
Total weight (kg.)	1290
Vehicle weight (kg.)	0
Towing weight (kg.)	0
Towing weight w/brakes (kg.)	0
No. of axles	2
License plate	AM51978
VIN	TMBZZAAZED631115
Date of first registration	2014-05-05
Status of registration	Afmeldt
Semiannual tax (DKK)	310,00 kr. (historically, license plates are deregistered)
Leasing period (if applicable)	2017-05-04 to 2017-07-04
Owner type	Privatperson
Status last updated	2017-07-04
Last inspection date	None
Last inspection result	
Debt in car?	False
Debt owed in car (DKK)	0
Creditors in car	
Insurance company	LÆRERSTANDENS BRAND
Insurance policy number	N/A
Insurance created	04-07-2017
Insurance status	Ophørt

Figure 5.13: The property view of a car with revoked plates. It has no insurance, is “afmeldt” (revoked) and we need to get the taxes from a json-list of historic payments instead of a field for the latest payment. This list is searched for the last full period (6 months), which were not a refund, and that amount is returned.

Property View	
— Properties	
Type	Car
Type of car	Motorcykel
Government vehicle?	
Allowed use of car	Privat personkørsel
Model	Harley Davidson - UOPLYST
Model year	
Odometer (km.)	0
Color	Ukendt
Fuel	Benzin
Fuel efficiency (km/l)	None
Engine power (hp)	
Engine cylinders	2
Top speed (km/h)	
Total weight (kg.)	425
Vehicle weight (kg.)	275
Towing weight (kg.)	None
Towing weight w/brakes (kg.)	None
No. of axles	2
License plate	ET11900
VIN	C1725
Date of first registration	1958-07-01
Status of registration	Registreret
Semiannual tax (DKK)	172,50 kr.
Leasing period (if applicable)	(not a leasing car)
Owner type	Privatperson
Status last updated	2006-04-28
Last inspection date	2006-01-09
Last inspection result	Godkendt
Debt in car?	False
Debt owed in car (DKK)	0
Creditors in car	
Insurance company	THISTED FORSIKRING G/S
Insurance policy number	N/A
Insurance created	28-04-2006
Insurance status	Aktiv

(a) Property view of a motorcycle.

Property View	
— Properties	
Type	Car
Type of car	Stor knallert
Government vehicle?	
Allowed use of car	Privat personkørsel
Model	Sun Trike 50
Model year	
Odometer (km.)	0
Color	Ukendt
Fuel	Benzin
Fuel efficiency (km/l)	None
Engine power (hp)	3
Engine cylinders	
Top speed (km/h)	
Total weight (kg.)	370
Vehicle weight (kg.)	182
Towing weight (kg.)	None
Towing weight w/brakes (kg.)	None
No. of axles	2
License plate	OJ109
VIN	9UATR0A1171900128
Date of first registration	2007-08-15
Status of registration	Registreret
Semiannual tax (DKK)	0 - no taxes are or have been imposed on this vehicle
Leasing period (if applicable)	(not a leasing car)
Owner type	Privatperson
Status last updated	2011-06-01
Last inspection date	2007-05-31
Last inspection result	Godkendt
Debt in car?	False
Debt owed in car (DKK)	0
Creditors in car	
Insurance company	PRIVATSIKRING
Insurance policy number	N/A
Insurance created	01-06-2011
Insurance status	Aktiv

(b) Property view of a scooter.

Figure 5.14: The property view of a MC and a scooter side-by-side.

Property View	
– Properties	
Type	Car
Type of car	Påhængsvogn
Government vehicle?	
Allowed use of car	Godstransport
Model	Va - 2000 KG
Model year	
Odometer (km.)	0
Color	
Fuel	None
Fuel efficiency (km/l)	None
Engine power (hp)	
Engine cylinders	
Top speed (km/h)	
Total weight (kg.)	2000
Vehicle weight (kg.)	0
Towing weight (kg.)	None
Towing weight w/brakes (kg.)	None
No. of axles	2
License plate	AH9433
VIN	UH72017H313358369
Date of first registration	2013-05-12
Status of registration	Registreret
Semiannual tax (DKK)	270,00 kr.
Leasing period (if applicable)	(not a leasing car)
Owner type	Privatperson
Status last updated	2013-05-12
Last inspection date	None
Last inspection result	
Debt in car?	False
Debt owed in car (DKK)	0
Creditors in car	
Insurance company	-
Insurance policy number	-
Insurance created	-
Insurance status	-

(a) Property view of a trailer.

Property View	
– Properties	
Type	Car
Type of car	Campingvogn
Government vehicle?	
Allowed use of car	Beboelse
Model	Knaus - SüDWIND 550
Model year	
Odometer (km.)	0
Color	
Fuel	Benzin
Fuel efficiency (km/l)	None
Engine power (hp)	
Engine cylinders	
Top speed (km/h)	
Total weight (kg.)	1500
Vehicle weight (kg.)	1200
Towing weight (kg.)	None
Towing weight w/brakes (kg.)	None
No. of axles	1
License plate	YJ2955
VIN	WKN55423J7W010288
Date of first registration	2009-07-22
Status of registration	Registreret
Semiannual tax (DKK)	707,00 kr.
Leasing period (if applicable)	(not a leasing car)
Owner type	Privatperson
Status last updated	2015-05-21
Last inspection date	2015-05-21
Last inspection result	Godkendt
Debt in car?	False
Debt owed in car (DKK)	0
Creditors in car	
Insurance company	-
Insurance policy number	-
Insurance created	-
Insurance status	-

(b) Property view of a caravan.

Figure 5.15: The property view of a trailer and a caravan side-by-side.

Property View	
— Properties	
Type	Car
Type of car	Personbil
Public/government vehicle?	Probably a police vehicle (it's self-insured)
Allowed use of car	Privat personkørsel
Model	Volkswagen Passat Variant 2.0 TDI BMT 177..
Model year	2014
Odometer (km.)	0
Color	Hvid
Fuel	Diesel
Fuel efficiency (km/l)	19.2
Engine power (hp)	130
Engine cylinders	4
Top speed (km/h)	220
Total weight (kg.)	2200
Vehicle weight (kg.)	0
Towing weight (kg.)	750
Towing weight w/brakes (kg.)	1800
No. of axles	2
License plate	AL45776
VIN	WVWZZZ3CZEE085369
Date of first registration	2014-03-25
Status of registration	Registreret
Semiannual tax (DKK)	890,00 kr.
Leasing period (if applicable)	(not a leasing car)
Owner type	Firma
Status last updated	2014-03-25
Last inspection date	None
Last inspection result	
Debt in car?	False
Debt owed in car (DKK)	0
Creditors in car	
Insurance company	SELVFORSIKRING
Insurance policy number	N/A
Insurance created	25-03-2014
Insurance status	Aktiv

(a) Property view of a police car. Notice in particular the insurance company saying “SELVFORSIKRING” (self-insurance) which to my knowledge is illegal in Denmark except for government entities. This may indicate government use and in particular a police car. We conclude on that in the upper field `Public/government vehicle?`.

Property View	
— Properties	
Type	Car
Type of car	Personbil
Public/government vehicle?	Probably (it doesn't pay taxes)
Allowed use of car	Ambulancekørsel
Model	Mercedes-Benz Sprinter 316 CDI
Model year	
Odometer (km.)	42000
Color	
Fuel	Diesel
Fuel efficiency (km/l)	10
Engine power (hp)	120
Engine cylinders	4
Top speed (km/h)	
Total weight (kg.)	3800
Vehicle weight (kg.)	3000
Towing weight (kg.)	None
Towing weight w/brakes (kg.)	None
No. of axles	2
License plate	AV89845
VIN	WDB9066331P135732
Date of first registration	2015-12-21
Status of registration	Registreret
Semiannual tax (DKK)	0 - no taxes are or have been impos
Leasing period (if applicable)	(not a leasing car)
Owner type	Firma
Status last updated	2015-12-21
Last inspection date	2017-01-26
Last inspection result	Godkendt
Debt in car?	False
Debt owed in car (DKK)	0
Creditors in car	
Insurance company	TRYG FORSIKRING A/S
Insurance policy number	N/A
Insurance created	21-12-2015
Insurance status	Aktiv

(b) Property view of an ambulance. This ambulance are not self-insured, but instead no taxes are paid on it, which otherwise only scooters are exempted from. We conclude that it must be a government vehicle and it is also registered as an ambulance.

Figure 5.16: The property view of a police car and an ambulance side-by-side to compare two types of government vehicles and the discrepancies they have compared to the other vehicle categories.

Property View	
— Properties	
Type	Car
Type of car	Lastbil
Public/government vehicle?	
Allowed use of car	Godstransport
Model	Scania - UOPLYST
Model year	
Odometer (km.)	0
Color	Ukendt
Fuel	Diesel
Fuel efficiency (km/l)	None
Engine power (hp)	
Engine cylinders	
Top speed (km/h)	
Total weight (kg.)	17500
Vehicle weight (kg.)	8500
Towing weight (kg.)	None
Towing weight w/brakes (kg.)	None
No. of axles	2
License plate	TY88978
VIN	XLEP4X20004388267
Date of first registration	1998-04-07
Status of registration	Afmeldt
Semiannual tax (DKK)	
Leasing period (if applicable)	(not a leasing car)
Owner type	Firma
Status last updated	2013-05-15
Last inspection date	2012-04-19
Last inspection result	Godkendt
Debt in car?	False
Debt owed in car (DKK)	0
Creditors in car	
Insurance company	CODAN
Insurance policy number	N/A
Insurance created	15-05-2013
Insurance status	Ophørt

(a) The property view of lorry (which also happens to have revoked plates).

Property View	
— Properties	
Type	Car
Type of car	Stor personbil
Public/government vehicle?	Probably (it doesn't pay taxes)
Allowed use of car	Kun godkendt til rutekørsel
Model	Volvo B7R UOPLYST
Model year	
Odometer (km.)	686000
Color	
Fuel	Diesel
Fuel efficiency (km/l)	None
Engine power (hp)	290
Engine cylinders	6
Top speed (km/h)	
Total weight (kg.)	18000
Vehicle weight (kg.)	11070
Towing weight (kg.)	None
Towing weight w/brakes (kg.)	None
No. of axles	2
License plate	BG88467
VIN	YV3R6R723BA143718
Date of first registration	2010-11-04
Status of registration	Registreret
Semiannual tax (DKK)	0 - no taxes are or have been imposed
Leasing period (if applicable)	(not a leasing car)
Owner type	Firma
Status last updated	2015-09-15
Last inspection date	2017-07-20
Last inspection result	Godkendt
Debt in car?	False
Debt owed in car (DKK)	0
Creditors in car	
Insurance company	Alm. Brand
Insurance policy number	N/A
Insurance created	29-12-2016
Insurance status	Aktiv

(b) Property view of a bus used for passenger services.

Figure 5.17: The property view of a lorry and a bus side-by-side.

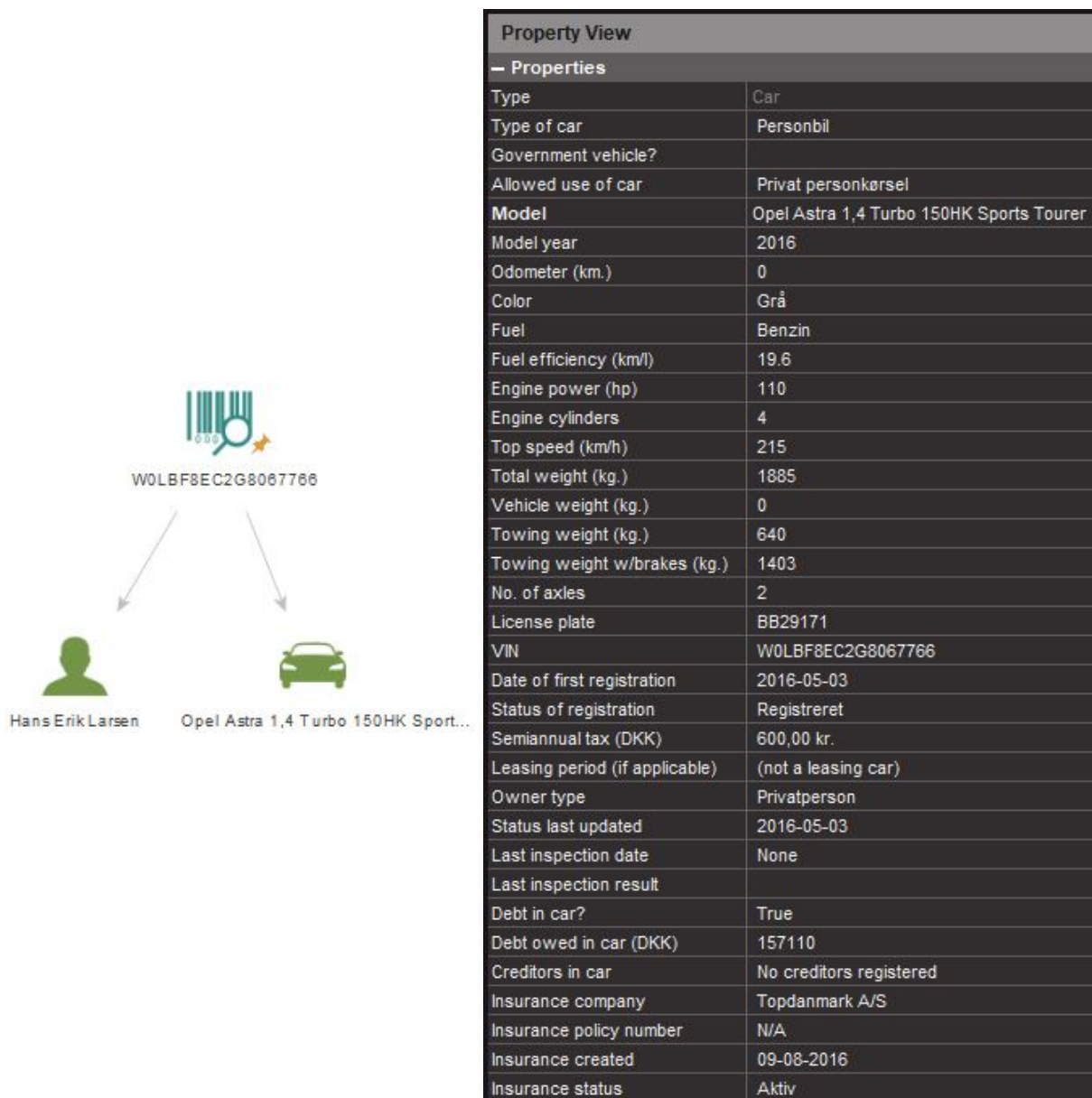


Figure 5.18: Graph and property view of a car found from running the transform on a VIN instead. Notice the odd case where debt is publicly notated with a debtor, but no creditors are returned.

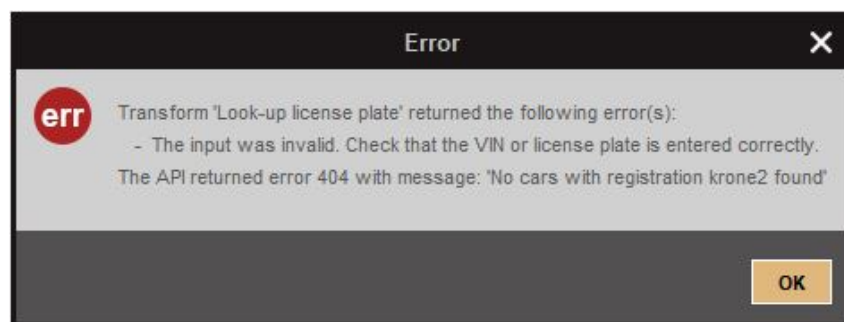


Figure 5.19: The error returned to the user if the license plate or VIN is invalid. We do not print the returned output in the console in this case, because all that is returned is put into the pop-up already.

5.2 Auto-generated report

In this section we verify that our report-generator program adheres to the requirements specifications put forward in Section 3.4.

The requirements covered both functional requirements of the report content and non-functional about the program's behavior (arguably also functional in their content as noted when setting them up). Verifying the requirements mostly non-functional (automation from Maltego export and user prompts) can be tested by inputting test-data, while the verification of the mostly functional requirements of the outputted report requires manually inspecting the pdf ensuring that statistics (incl. figures), scenarios and standards are displayed properly.

We employ a "white-box"-approach to avoid repeating tests. We know that the code generating the sections on scenarios and standards are virtually identical and that successfully generating one section with the subsequent listings of data are independent from the number of findings/data used as input. In the current implementation is sufficient to input only one finding and assigning it many labels to verify that some input from Maltego can successfully generate the required elements in the report.

For testing the mostly non-functional requirements, we sample a Maltego-export containing a selection of output that can be expected from a Maltego-investigation. An example of such a sample is given in Appendix D.2.

Running the program we get presented with a prompt, select the exported sample and get presented with the multichoice-prompts shown in Figure 4.5 as expected. As can also be seen on the figure, the labels are identical or close to those on the mindmap (Fig. 3.1). We should also test that prompts allow for none, one and several selected options; this is included in the functional tests.

The following simple test verifies that the outer working of the program *satisfies* the non-functional requirements of working with the Maltego-export and offering a simple labeling of input.

Verifying that the content is correctly generated however requires manual inspection of the output pdf.

The goal is to test the different functions of the dynamic content in the report: Frequency diagram, lamp (red, yellow, green), tables (including coloring), listing of satisfied scenarios/standards and their findings in each section. In essence it only requires one line in the input csv to test this: By selecting a lot of labels for the single finding, it will simply appear all over the report; we know that the program currently does not distinguish between one or several findings for a requirement to be satisfied.

Generation of all content apart from the lamp is independent of the number of labels assigned. The frequency diagram and -tables do show a count, but it is generated disregarding how many labels or findings are in the input. Similar for listing of findings per scenario and whether they are satisfied, it is independent from the number of findings satisfying that scenario.

Following this, we need to test that:

1. We can generate a green, yellow and red lamp and that the executive summary is generated appropriately.
2. The statistics section reflects the chosen labels per finding (including the frequency diagram).
3. The scenario section reflects the chosen labels per finding per scenario, correctly shows satisfied requirements per scenario and overall and has the expected explanations of each scenario.
4. The standards section reflects the chosen labels per finding per standard and correctly shows violated requirements per standard and overall.

We run three tests to verify the first claim. The results are seen in Figure 5.20 showing the dynamically generated executive summary. The report were generated using two findings in this case.

The frontpage is generated in an identical way with the lamp and client name used here (see an example of a full report in App. B.3).

The executive summary is easy to interpret and lists a simple, short summary of the findings. Using the lamp this information is also conveyed clearly. In a short disclaimer, we highlight the conditions under which the report is made and can be used.

It thus fits the criteria and **pass** the test.

To ensure that the following three claims are fulfilled, we take the first 10 entries of the sample output from Maltego (as seen in App. D.2) and enter them in the program, labeling the appropriately as a user might find it relevant to do.

The output of this is a complete report, which is very long and thus only shown in Appendix B.3. To compare, we also generate a report run on the same 10 entry long `csv`-file, but now only assigning one label for one of the findings. The resulting report is found in Appendix B.4.

We can now observe the dynamic generated sections.

If we look at the content in the statistics-section called *“Data found”*, we observe the frequency diagram and subsequent sections on each primary data category, where the labels used are given with their respective frequency. Only assigned labels are shown, because the opposite would the disturb the picture.

The *“Scenarios”*-section is introduced by a short motivation and explanation of the origin of the chosen attack scenarios as well as a disclaimer detailing how the results can be applied. In an easy-to-read table below, the scenarios are listed with their title and using colors, we show if the

client is considered vulnerable to each of them. We see that the `true/false` are generated for scenario and colored.

In the subsequent sections, each scenario is explained in detail to convey an understanding of how the requirements of the scenario can enable the scenario and with some examples. The goal was to inform of patterns and mediate awareness and we see this is done (using the explanations given in Section 2.5) for each of the scenarios. Trailing the tabular overview (also easy to decode using colors and a simple `true/false`-statement) of each scenario is a listing of the scenario's requirements. Here the findings put into the program are shown (as labeled), further enabling the reader to understand how the findings are linked to the scenario.

From the explanation given in the introduction to each scenario, all together this should awake some thinking in the reader to enable him to grasp a better understanding of each scenario and be able to recognize them individually and in combination (which we do note to the reader is a plausible thing to happen).

The “*Standards*”-section pretty much follows the same pattern. We see the dynamically generated introduction to both the main section and the subsections using the same easy-to-read colored tables and introductions with numbering consistent with the input to the program. The introduction also contains the necessary explanation of the standards considered and a disclaimer about their application.

In summary we have shown that the automatically generated report fulfills the requirements as formulated in the analysis in Section 3.4.

In particular we have demonstrated the ability to generate a statistical summary of findings from an OSINT-gathering, where a frequency diagram and -tables summarizes the findings within five categories enabling to compare categories and identify problematic areas.

We also successfully generated and related findings to select, common targeted OSINT-enabled scenarios with explanations and examples, while emphasizing the difference that may exist between theory and practice to the reader.

We connected the findings with a smaller subset of the applicable legislation, standards and guidelines of Section 2.3. It was possible to relate the findings directly to some policies/controls of them, which demonstrates the viability of this approach. With further work it is possible to link the labels of the findings with policies/controls of all relevant legislation, standards and guidelines and give the reader insights to a domain otherwise somewhat disconnected from actual findings.

Through out the report, we employed easy-to-digest figures and tables with colors for immediate interpretation. Together with the executive summary and its “lamp”, the report is accessible for people with different backgrounds to read it as they desire – in full or just a summary.

With the relevant improvements regarding linkage of labels to scenarios and standards implemented, it can thus be included in the work routine or portfolio of a security consultancy as applicable to their needs (either as an alternative view on findings or as a independent delivery to the client).

We can conclude that the program **satisfies** the tests and hence the requirements.

1 Executive summary

Based on this report detailing the findings of an *Open Source Intelligence* gathering performed on ACME A/S, it is found that **ACME A/S is vulnerable to 0 of 5 common, OSINT-enabled cyber attack scenarios** reviewed and **violates 0 of 8 standards and guidelines**, which are expected to be applicable to ACME A/S as an organization operating in Denmark. This results in ● severity, which is a good result!

2 findings from the OSINT-gathering were considered for this report.

The conclusions in this report is drawn from a number of commonly occurring scenarios and standards used and may not apply to ACME A/S directly. The results should be considered in a larger context with respect to the overall security maturity of ACME A/S and the risk appetite. Instead the results can be used to – in a simple way – understand the context in which the findings of the OSINT-gathering resides and enhance the understanding and procedures around OSINT-data and its influence on ACME A/S in daily business operations.

1 Executive summary

Based on this report detailing the findings of an *Open Source Intelligence* gathering performed on ACME A/S, it is found that **ACME A/S is vulnerable to 1 of 5 common, OSINT-enabled cyber attack scenarios** reviewed and **violates 3 of 8 standards and guidelines**, which are expected to be applicable to ACME A/S as an organization operating in Denmark. This results in ● severity, which is satisfactory, but with room for improvement.

2 findings from the OSINT-gathering were considered for this report.

The conclusions in this report is drawn from a number of commonly occurring scenarios and standards used and may not apply to ACME A/S directly. The results should be considered in a larger context with respect to the overall security maturity of ACME A/S and the risk appetite. Instead the results can be used to – in a simple way – understand the context in which the findings of the OSINT-gathering resides and enhance the understanding and procedures around OSINT-data and its influence on ACME A/S in daily business operations.

1 Executive summary

Based on this report detailing the findings of an *Open Source Intelligence* gathering performed on ACME A/S, it is found that **ACME A/S is vulnerable to 5 of 5 common, OSINT-enabled cyber attack scenarios** reviewed and **violates 3 of 8 standards and guidelines**, which are expected to be applicable to ACME A/S as an organization operating in Denmark. This results in ● severity, which is not good!

2 findings from the OSINT-gathering were considered for this report.

The conclusions in this report is drawn from a number of commonly occurring scenarios and standards used and may not apply to ACME A/S directly. The results should be considered in a larger context with respect to the overall security maturity of ACME A/S and the risk appetite. Instead the results can be used to – in a simple way – understand the context in which the findings of the OSINT-gathering resides and enhance the understanding and procedures around OSINT-data and its influence on ACME A/S in daily business operations.

Figure 5.20: The dynamically generated content of the executive summary adjusting to the findings (highlighted with bold). Notice the use of small text insert to conclude on the severity as well as the client name, which is also parsed as an argument in a similar way.

Chapter 6

Discussion

The subject of this thesis came to birth from a discussion between IT security professionals on the issue of the mass of OSINT-data generated daily about us as individuals and organizations. In particular we discussed methods enabling an enhanced data collection and reporting to gain an overview of it. It was asserted that it could be possible to automate some, if not all, parts of this task and create regularly scheduled automated reporting for businesses. However none of us knew of the exact implications of this, which this thesis came to explore.

It would have been interesting to perform more than the report only automatically; before the project began, I had hoped to be able to automate some of the work around the transforms on Maltego. Some components exist to help automate Maltego transforms, but they are inadequate. Apart from some simple macros that can be created within Maltego (see App. A.1), this is not possible. I had hoped simple coding or a simple AI could have interacted with the transforms or Maltego. The AI would however prove difficult, as the reconnaissance does not have a *final* goal as such and thus no baseline for when to stop execution.

Doing reconnaissance is a dynamic assignment, which depends heavily on the target and the result found during the scan which in turn may influence the analyst's next choice. As soon as the intelligence gathering is not merely "look up some information on service X", the researcher always has to perform some manual thinking to get the intelligence and combine the results.

The tools might very well only be the "simple" ones from Section 2.1.2.2, but with the added knowledge of an experienced researcher, he can deduce new meaning and get different angles on previous findings. He will also add knowledge of scenarios to determine which step to take next with the information currently at hand – this requires the flexible thinking of a human mind to alter scenarios ever-so slightly from "the portfolio" of known scenarios to create new ones.

It is this "dynamic" nature of the pen-test, which makes the automation of it quite difficult and probably something that cannot be solved adequately without AI; *"Judgment is required in selecting the appropriate tools and in identifying attack vectors that typically cannot be identified through automated means."* [45].

Especially if we move towards HUMINT, great responses to some specific action of the target

can be prepared and coded (or handled by an AI), but many conversations can take unforeseen turns or in many cases, require human interaction not able to be automated currently.

In Chapter 2 we looked into an extensive amount of standards and guidelines worthy of an entire literature study by itself. We had to scope to only search in sources of known credibility. Despite the amount of content, most with sound advice for enhancing overall security maturity in organizations, next to nothing where relevant to *outbound* information sharing, the type of communication leaving a trail of OSINT on the user. Further studies and input from several sources with practical experience could be used to establish convincing baselines and best practices.

Theory-wise our survey showed that it is an area of little recognition, which both speaks to the necessity of a product like this, but also complicates the creation of it. It may be a viable approach to conduct qualitative studies on the subject by interviewing professionals to give input (especially ones with varying backgrounds). The theory in Section 2.3 should surely form a starting point for this work.

Establishing the scenarios to be linked to the findings was simpler, but also a much more subjective area. It was asserted that common threat analysis could provide a framework for it, but the models surveyed are not “catch-all” in the sense they cannot work well without a specific target to view the threats in context of.

The goal was to list “common OSINT-enabled cyber attack scenarios” and we did so by referring to recent news articles and descriptions of actual attacks – not an entirely unviable approach, but it lacks a standardized methodology, prevalent to this area of expertise in general. The danger is that the scenarios suffer from e.g. confirmation bias in terms of attack capabilities, preferred methods, OSINT requirements, which can make the conclusions of the following auto-generated report misleading.

It is important to have in mind, that it is an attempt to distill the workings of people that pride themselves in thinking out-of-the-box; it is not entirely an easy target to catch up with in scientific writing.

We have been able to produce both software enhancing the search of OSINT-data as well as a framework for automating conclusion on the findings. We find that both deliveries are adhering to the requirements put forward in Chapter 3, but both were more time-consuming than expected thus kept on a level of proof-of-concept.

The two integrations developed for the Danish domain register and the Danish license plate registry as transforms for Maltego are simple (as opposed to the report-generation’s integration with both standards and legislation) yet powerful and can move forward to practical use as-is. They enable the user to acquire OSINT on Danish domains and vehicles respectively and do so in an easy, standardized manner. This is a feature that can be encompassed into the work routine of e.g. the intelligence gathering-phase of a pen-test (see Sec. 2.1) or independent investigations on the issue of exposure to OSINT-enabled attacks (e.g. vulnerability assessments).

The transform-development should be pursued further. The sources were chosen to integrate data from based on the requirements. Following the considerations of the often practice-oriented approach of security researchers, one could also here conduct interviews to establish a prioritized list of integrations to develop.

Connecting and relating the findings of the intelligence gathering-phase to actual attack scenarios in the framework for auto-generating the report proved more difficult than predicted.

It was expected that one could work from categories of data labels and tie them with both scenarios and standards in a straight-forward manner. The differences in the domains of the three were however quite wide, resulting in such discrepancies that the approach came to just link them for the sake of example and to be able to generate a report all-together.

The standards and attack scenarios comes from two different domains; there is a difference in perspective and granularity with the scenarios aiming to explain events and the standards policies, controls and managerial course of actions. The data categories are made from a software engineer's perspective and atomized by subject within the field, which do not necessarily match the two others (standards and attack scenarios). In addition, the standards/guidelines often rely on knowing the circumstances the data appeared under (e.g. found on the organization's website or a 3rd party).

In regards to the first requirement to the framework ("enable identification of areas in need of mitigative steps"), this may be improved with a different categorization; we succeeded to do that, however it is easier if the *origin* of the findings are known so one can identify from where leaks occurred. This can also offer a different approach for linking scenarios/standards with findings, as the scenarios can benefit from having a context to the data found and thus differentiate between e.g. data found in a context controlled by the client as opposed to a third party.

Similarly the methods of vulnerability testing mentioned in Section 2.1.4 may offer ideas.

An alternative approach could be to establish the scenarios or the relevant standards first, and from them create the data labels (the opposite of the current approach). On the other hand, this may lead one to consider data labels too narrowly – especially if the scenarios are not broad enough.

The focus should be on making all three categories form one, coherent mesh of interrelated data categories and requirements. It will also improve the amount of information that the report can relay to readers, as it will make the report's conclusions clearer by following the flow of findings over data labels to scenarios and standards. We can do it now, but arguably a person with insight into information mediation can enhance it further. Other examples of reports should be considered; there may also be more lessons to learn from the two reports considered in this thesis. In Section 3.4.1 we offer even two more options to redesign the data labels in an alternative way.

There are so many little pieces of information to be discovered included in business processes in varying number of ways; connecting them all correctly to applicable scenarios to the specific organization, is not possible to do in a homogeneous and meaningful way across different orga-

nizations. It was pointed out in the introduction (Sec. 1) and [35], that some information are directly to be considered “confidential”, while much of the rest are of a “semi-confidential” status, where the company may keep it to themselves, but do not consider the knowledge or possession of the data to be a breach of confidentiality.

Capturing this in the data classification and later in the link between data and scenarios, will have discrepancies between organizations (security consultancy clients), as this “semi-confidential” data is specific to organizational culture and configuration.

The standards were sufficiently broad in their wording, that this problem did not arise, although the information always had to be seen in relation with the organization’s current setup/processes, as some information could be considered e.g. confidential between suppliers in one place, but not the next.

We could conclude the deliveries were viable solutions to the problem put forward in Section 1.1. Such a conclusion will however always rely on the test cases used.

As noted in Section 2.1, different researchers have different approaches for performing their intelligence gathering. I am only one to perform the investigation from which the test were built, were as another, more experienced researcher may come out with entirely different findings.

There is also some element of confirmation bias (cf. the section on psychology, Sec. 2.4) in the investigation performed here, which is inherent to all researchers performing a pen-test or an intelligence gathering: They will have a preconceived idea of what to find and weigh findings and patterns confirming these higher. This can prove to be an element which overall will make such a product as this, hard to be accepted as common tool between security researchers (as otherwise intended), because their bias consistently tells them different from my bias used to create and link the scenarios.

On the other hand, the transforms are a free of any bias and just return all the information available. This allows a researcher to freely follow any direction he believes most interesting or valuable to his current scenario, making it a better product in a field of many individualists.

Chapter 7

Conclusion

The aim of this thesis was to develop tools for enhancing the identification, collection and reporting of OSINT on Danish organizations. Emphasis was put on the tools being applicable to security professionals working with this on a regular basis as well as being easy to use and interpret. For this, the current state-of-the-art methodologies for performing analysis of this were examined together with legislation, standards and guidelines applying to the area.

The solution brought forward were Maltego transforms for Danish OSINT-sources, as well as a framework for automatically generating a report of the findings in an OSINT-investigation and relating them with the applicable legislation/standards as well as common OSINT-enabled cyber attack scenarios.

Specifically we made a proof-of-concept implementation of the DK Hostmaster API and the registry of Danish vehicles as transforms for the widely used OSINT-investigation platform Maltego. Through extensive testing, the transforms were found to adhere to the design guidelines for Maltego transforms and our requirements of improving the information gathering process with an easy-to-use tool providing additional insights. This allows for the transforms to be published to the integrated marketplace of transforms in Maltego and thus a commercial application. The transforms provides an easy front to interact with the respective API's, which is a novel approach for Danish OSINT-sources, which are not included before in such tools as Maltego.

If the tools developed are put into use, they have the potential to reduce the workload associated with collection of OSINT-data significantly. Maltego already provides many basic transforms; to further be able to employ the many Danish OSINT-sources here is a great advantage to the user. Without them, details such as contact information of domain owners and debt-information, which we can now provide, would have gone unnoticed, disabling the researcher from getting the full picture and draw correct conclusions on Danish organization's exposure to OSINT-enabled attacks.

These conclusions are aided by the automatically generated report which the framework can generate based on the Maltego export. The report attempts a novel approach to link the findings from Maltego with a subset of applicable scenarios, standards and guidelines.

The reader (either a security consultant or the client) thus gets insight to scenarios, the prerequisites of them and how findings from the investigation relates hereto. The improved insight can in turn raise the general level of awareness among the reader such that he himself are able to mediate it to his peers.

The work on the report generation framework and the prior analysis however highlighted the difficulties connecting the domains of legislation and standards with practical attack scenarios and concrete findings. The three domains are very different and will require input from several sources to form a sufficiently coherent mesh. We have suggested reasons for this and highlighted alternative approaches, which we invite others to pursue based on the preliminary descriptions of the state-of-the-art methods and tools used within the field of cyber security found here (specific suggestions for future work are found in Section 7.1).

This area was identified as a risk at project initiation as per Section 1.1.2. These risk uncovered areas to begin the initial work. In particular we dedicated a lot of time to examine sources of standards, guidelines and applicable Danish legislation to gain an understanding of what content was available. This work was a necessity to continue to develop a body of the auto-generated report. Only after it was finished, where we able to try to establish how the report could be generated. It was here where we discovered that by a combination of too little literature on controlling OSINT-data and an output containing less information than expected from Maltego, automatically generating a report became more difficult. In addition, the process to link the findings with the legislation and standards required much more work than expected as the connections could not be made in the straight-forward manner originally asserted. This was not considered a risk at project start and thus only researched some time into the project work.

We did however succeed to develop the framework. While there is work to be done on the logic on the linking, processing of the best possible data export from Maltego can be done and a result much like professional reports generated.

Conclusively, both deliveries are in their current state usable and adhere to the analyzed specifications. As proof-of-concepts they demonstrate a viable two-piece solution to enhance identification and reporting of OSINT-data on Danish organizations. A set of transforms (with the two developed included as-is) can be used for performing OSINT-gathering and with further work on the underlying data model of legislation/standards, scenarios and data labels, it is possible automatically generate a report concluding on such an information gathering to benefit the busy employees responsible for organizational security in the receiving end.

7.1 Future work

To utilize the work done here, we suggest the following areas of consideration for future work to be done on the deliveries of this report to support the overall goal to continue to enhance the identification and reporting of OSINT on Danish organizations:

- Consider the layout of the auto-generated report if the reader is proficient within the field of design, layout or (scientific) communication. The current content only contains basic

considerations as expected from an engineer's approach. Section 2.1.6 contains some advice that are not implemented like showing the most severe violations first.

- Consider a “slimmer” version of the auto-generated report by e.g. removing scenarios or standards not present.
- The auto-generated report could be extended with explanations of the standards. To some extent the labels and scenarios can also be explained more elaborate, but we did not deem this necessary.
- Investigate a different approach to link scenarios and standards to findings. In Section 6 we suggest two approaches and in Section 3.4.1 we provide two others, all which may be worthwhile to explore.
- Continue the development of the transforms and mature them for commercial distribution. As per the design guidelines for the transforms this will require documentation for transforms, as well as licensing needed to be taken into consideration. Use input (focus groups?) from professionals. The cost of server space at Paterva (necessary for having the transform published on the transform hub) and pricing need to be identified in connection to this; it may be a deciding factor if it is possible to move forward and offering them as a service.
- Prepare the product for sale as a part of a IT security company's portfolio of services. This can include evaluation of usability, suggestions for additional data sources, qualitative research on the requirements of CISO's, value propositions in a current, commercial context and a project presentation – all in collaboration with some IT security company's consultants and marketing departments. Eventually they can be tested in real-life customer meetings to evaluate the performance.

References

- [1] S. Ahmet and B. Yener. Modeling and Detection of Complex Attack. *SecureComm*, pages 234–243, 2003. URL <https://pdfs.semanticscholar.org/20d8/739b6de908d97ac069e87f411b8ebaf709be.pdf>.
- [2] Balabit Corp. Social Engineering Leads the TOP 10 List of Most Popular Hacking Methods – Balabit Survey Results from Black Hat USA and EU shows, 2016. URL <https://www.balabit.com/news/press/social-engineering-leads-the-top-10-list-of-most-popular-hacking-methods-balabit-survey-results-from-black-hat-usa>.
- [3] Joseph Bonneau, Elie Bursztein, Ilan Caron, Rob Jackson, and Mike Williamson. Secrets, Lies, and Account Recovery: Lessons from the Use of Personal Knowledge Questions at Google. doi: 10.1145/2736277.2741691. URL <http://dx.doi.org/10.1145/2736277.2741691>.
- [4] Pauline Bowen, Joan Hash, and Mark Wilson. NIST 800-100: Information Security Handbook: A Guide for Managers. Technical Report October, NIST, 2006. URL <http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-100.pdf>.
- [5] Dawn Cappelli, Michelle Keeney, Eileen Kowalski, Andrew Moore, and Marisa Randazzo. Insider Threat Study: Illicit Cyber Activity in the Banking and Finance Sector (presentation), 2005. URL <https://www.cs.auckland.ac.nz/courses/compsci725s2c/archive/2005/lectures/presentations/mz.pdf>.
- [6] Richard A. Caralli, James F. Stevens, Lisa R. Young, and William R. Wilson. Introducing OCTAVE Allegro : Improving the Information Security Risk Assessment Process. *Young*, (May):154, 2007.
- [7] Center for Cybersikkerhed. Phishing uden fangst - Udenrigsministeriet under angreb (undersøgelsesrapport). Technical report, CFCS, 2016. URL <https://fe-ddis.dk/cfcs/cfcsdocuments/phishingudenfangst.pdf>.
- [8] Center for Cybersikkerhed. Spear-phishing – et voksende problem. Technical report, 2016. URL <https://fe-ddis.dk/cfcs/CFCSDocuments/Spearphishingsikkerhedsanbefaling.pdf>.
- [9] Center for Cybersikkerhed. Trusselsvurdering: Cybertruslen mod telesektoren i Danmark (februar 2017). Technical report, Center for Cybersikkerhed, 2017.

- [10] Centre for the Protection of National Infrastructure. Secure Online Presence. URL <https://www.cpni.gov.uk/secure-online-presence>.
- [11] Centre for the Protection of National Infrastructure. Understanding and Countering The Phishing Threat. page 11, 2016. URL <https://www.cpni.gov.uk/system/files/documents/23/de/understanding-hostile-reconnaissance-understanding-and-countering-the-threat.pdf>.
- [12] Larry Clavette, Capt. David Faggard, Paul F. Bove, and Joseph S. Fordham. New Media and The Air Force. Technical report, US Air Force, 2009.
- [13] Dansk Standard. DS/ISO/IEC 27005 – Information security risk management. Technical report, Danish Standards Association, 2011.
- [14] Dansk Standard. DS/EN ISO/IEC 27000:2017 – Information security management systems – Overview and vocabulary. Technical report, Danish Standards Association, 2017.
- [15] Dansk Standard. DS/EN ISO/IEC 27001:2017 – Information security management systems – Requirements. Technical report, Danish Standards Association, 2017.
- [16] Dansk Standard. DS/EN ISO/IEC 27002:2017 – Code of practice for information security controls. Technical report, Danish Standards Association, 2017.
- [17] Dansk Standard. DS/ISO/IEC 27004:2016 – Information security management – Measurement. Technical report, Danish Standards Association, 2017.
- [18] Danny Dhillon. Developer-driven threat modeling: Lessons learned in the trenches. *IEEE Security and Privacy*, 9(4):41–47, 2011. ISSN 15407993. doi: 10.1109/MSP.2011.47.
- [19] Digitaliseringsstyrelsen. Eksterne krav til informationssikkerhed, 2016. URL <https://www.digst.dk/Informationssikkerhed/Implementering-af-ISO27001/Eksterne-krav-til-informationssikkerhed>.
- [20] Digitaliseringsstyrelsen and Center for Cybersikkerhed. Cyberforsvar der virker. Technical report, 2017. URL https://fe-ddis.dk/cfcs/publikationer/Documents/Cyberforsvardervirker-2017_110117.pdf.
- [21] Golnaz Elahi and Eric Yu. A Goal Oriented Approach for Modeling and Analyzing Security Trade-Offs. (July):375–390, 2007. doi: 10.1007/978-3-540-75563-0_26.
- [22] Martin Fowler and Kendall Scott. *UML Distilled – Applying the Standard Object Modeling Language*. Addison-Wesley, 2 edition, 1997. ISBN 0201325632.
- [23] Philip Hunter. Social networking: the focus for new threats - and old ones. *Computer Fraud and Security*, 2008(7):17–18, 2008. ISSN 13613723. doi: 10.1016/S1361-3723(08)70114-3.

- [24] Eric M. Hutchins, Michael J. Cloppert, and Rohan M. Amin. Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains. Technical report, Lockheed Martin Corporation, 2011. URL <http://www.lockheedmartin.com/content/dam/lockheed/data/corporate/documents/LM-White-Paper-Intel-Driven-Defense.pdf>.
- [25] Information Security and Identity Management Committee (ISIMC), Network and Infrastructure Security Subcommittee (NISSC), and Web 2.0 Security Working Group (W20SWG). Guidelines for Secure Use of Social Media by Federal Departments and Agencies (vers. 1.0). Technical Report September, Federal CIO Council, 2009. URL https://cio.gov/wp-content/uploads/downloads/2012/09/Guidelines_for_Secure_Use_Social_Media_v01-0.pdf.
- [26] Kyle Ingols, Matthew Chu, Richard Lippmann, Seth Webster, and Stephen Boyer. Modeling modern network attacks and countermeasures using attack graphs. *Proceedings - Annual Computer Security Applications Conference, ACSAC*, pages 117–126, 2009. ISSN 10639527. doi: 10.1109/ACSAC.2009.21.
- [27] Interpol. Social engineering fraud: questions and answers, 2015. URL <https://www.interpol.int/Media/Files/Crime-areas/Financial-crime/Social-engineering-fraud/>.
- [28] Daniel Kahneman. *Thinking, Fast and Slow*. Penguin Books, 2011. ISBN 9780141033570.
- [29] Michelle Keeney, Eileen Kowalski, Dawn Cappelli, Andrew Moore, Timothy Shimeall, and Stephanie Rogers. Insider Threat Study: Computer System Sabotage in Critical Infrastructure Sectors. *U.S. Secret Service and CERT Coordination Center/SEI*, (May):1–44, 2005. URL <http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Insider+Threat+Study:+Computer+System+Sabotage+in+Critical+Infrastructure+Sectors#0>.
- [30] Peter Kim. *The Hacker Playbook 2: Practical Guide To Penetration Testing*. CreateSpace Independent Publishing Platform, 2015. ISBN 9781512214567.
- [31] Lin Liu, Eric Yu, and John Mylopoulos. Security and Privacy Requirements Analysis within a Social Setting. *Proceedings 11th IEEE International Requirements Engineering Conference 2003*, 3:151–161, 2003. ISSN 1090-705X. doi: 10.1109/ICRE.2003.1232746. URL <http://portal.acm.org/citation.cfm?id=943910>.
- [32] Mass Soldal Lund, Bjørnar Solhaug, and Ketil Stølen. Model-driven risk analysis: The CORAS approach. *Model-Driven Risk Analysis: The CORAS Approach*, pages 23–43, 2011. doi: 10.1007/978-3-642-12323-8.
- [33] Pratyusa K. Manadhata and Jeannette M. Wing. An attack surface metric. *IEEE Transactions on Software Engineering*, 37(3):371–386, 2011. ISSN 00985589. doi: 10.1109/TSE.2010.60.

- [34] J. D. Meier, Alex Mackman, Michael Dunner, Srinath Vasireddy, Ray Escamilla, and Anandha Murukan. Improving Web Application Security: Threats and Countermeasures, 2003. URL <https://msdn.microsoft.com/en-us/library/aa302419.aspx>.
- [35] Kevin David Mitnick and William L. Simon. *The art of deception*. Wiley Publishing, Inc., 2002. ISBN 076454280X.
- [36] Suvda Myagmar, Adam J. Lee, and William Yurcik. Threat Modeling as a Basis for Security Requirements. In *StorageSS '05: Proceedings of the 2005 ACM workshop on Storage security and survivability*, pages 94–102, 2005.
- [37] National Cyber Security Centre. Common Cyber Attacks: Reducing The Impact. Technical Report January, GCHQ, 2016. URL https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/400106/Common_Cyber_Attacks-Reducing_The_Impact.pdf.
- [38] National Cyber Security Centre. Whaling: how it works, and what your organisation can do about it, 2016. URL <https://www.ncsc.gov.uk/guidance/whaling-how-it-works-and-what-your-organisation-can-do-about-it>.
- [39] National Cyber Security Centre. 10 steps to cyber security – Executive summary, 2017. URL <https://www.ncsc.gov.uk/guidance/10-steps-cyber-security>.
- [40] Ebenezer A Oladimeji, Sam Supakkul, and Lawrence Chung. Security threat modeling and analysis: A goal-oriented approach. *Proc of the 10th IASTED International Conference on Software Engineering and Applications SEA 2006*, pages 13–15, 2006. ISSN 0889866422. URL [https://pipeline.utdallas.edu/\\$\sim\\$eao015100/documents/SecurityThreatModeling.pdf](https://pipeline.utdallas.edu/\simeao015100/documents/SecurityThreatModeling.pdf).
- [41] Marwan Omar. *Insider threats: Detecting and controlling malicious insiders*. PhD thesis, Nawroz University, Iraq.
- [42] Andreas L. Opdahl and Guttorm Sindre. Experimental comparison of attack trees and misuse cases for security threat identification. *Information and Software Technology*, 51(5): 916–932, 2009. ISSN 09505849. doi: 10.1016/j.infsof.2008.05.013. URL <http://dx.doi.org/10.1016/j.infsof.2008.05.013>.
- [43] Paterva. Maltego Documentation: Custom Entities, . URL <https://docs.paterva.com/en/developer-portal/custom-entities/>.
- [44] Paterva. Maltego Documentation: Transform Hub Guidelines, . URL <https://docs.paterva.com/en/developer-portal/transform-hub-guidelines/>.
- [45] PCi Security Standards Council. Penetration Testing Guidance. Technical Report 1.0, PCi, 2015. URL https://www.pcisecuritystandards.org/documents/Penetration_Testing_Guidance_March_2015.pdf.

- [46] Proofpoint Inc. The Human Factor 2016. Technical report, 2015. URL <https://www.proofpoint.com/sites/default/files/human-factor-report-2016.pdf>.
- [47] Marisa Randazzo, Michelle Keeney, Eileen Kowalski, Dawn Cappelli, and Andrew Moore. Insider threat study: Illicit cyber activity in the banking and finance sector. *Finance*, 38(August):3–14, 2005. ISSN 07321872. doi: 10.1080/07321870590933292. URL <http://stinet.dtic.mil/oai/oai?&verb=getRecord&metadataPrefix=html&identifier=ADA441249%5Cnhttp://oai.dtic.mil/oai/oai?verb=getRecord&metadataPrefix=html&identifier=ADA441249>.
- [48] Vineet Saini, Qiang Duan, and V Paruchuri. Threat modeling using attack trees. *Journal of Computing Sciences in Colleges*, 23(4):124–131, 2008. ISSN 1937-4771. URL <http://dl.acm.org/citation.cfm?id=1352100>.
- [49] Karen Scarfone, Murugiah Souppaya, Amanda Cody, and Angela Orebaugh. NIST 800-115: Technical Guide to Information Security Testing and Assessment. Technical report, NIST, 2008. URL <http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-115.pdf>.
- [50] Bruce Schneier. Attack Trees. *Dr. Dobb's Journal*, 1999. URL <http://www.drdobbs.com/attack-trees/184411129>.
- [51] Adam Shostack. Experiences threat modeling at Microsoft. *CEUR Workshop Proceedings*, 413:1–11, 2008. ISSN 16130073.
- [52] Guttorm Sindre and Andreas L. Opdahl. Eliciting security requirements with misuse cases. *Requirements Engineering*, 10(1):34–44, 2005. ISSN 09473602. doi: 10.1007/s00766-004-0194-4.
- [53] Robert D. Steele. Open Source Intelligence: What Is It? Why Is It Important to the Military? In *Open Source Intelligence: READER Proceedings, 6th International Conference and Exhibit Global Security and Global Comp*, number Vol. 2, pages 329–341. Open Source Solutions, Inc., 1997.
- [54] Joint task force transformation initiative. NIST 800-30r1: Guide for conducting risk assessments. Technical Report September, NIST, 2012. URL <http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-30r1.pdf>.
- [55] Joint task force transformation initiative. NIST 800-53r4: Security and Privacy Controls for Federal Information Systems and Organizations Security and Privacy Controls for Federal Information Systems and Organizations. Technical report, NIST, 2014. URL <http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-53r4.pdf>.
- [56] The Penetration Testing Execution Standard Group. The Penetration Testing Execution Standard, 2009. URL http://www.pentest-standard.org/index.php/Main_Page.

- [57] T Tidwell, R Larson, K Fitch, and J Hale. Modeling Internet Attacks. In *Proceedings of the 2001 IEEE*, number June 2001, pages 54–59, 2001. ISBN 0780398149. URL <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.108.9040&rep=rep1&type=pdf>.
- [58] Anita Vorster and Les Labuschagne. A framework for comparing different information security risk analysis methodologies. *Information Security*, 193(C):95–103, 2005. URL <http://portal.acm.org/citation.cfm?id=1145686>.
- [59] Chan Tuck Wai. InfoSec Reading Room: Conducting a Penetration Test on an Organization. Technical report, SANS Institute, 2002. URL <http://www.sans.org/reading-room/whitepapers/auditing/conducting-penetration-test-organization-67>.
- [60] Gregory M. Walton, Geoffrey L. Cohen, David Cwir, and Steven J. Spencer. Mere belonging: The power of social connections. *Journal of Personality and Social Psychology*, 102(3):513–532, 2012. doi: 10.1037/a0025731. URL <http://psycnet.apa.org/?&fa=main.doiLanding&doi=10.1037/a0025731>.
- [61] Georgia Weidman. *Penetration Testing: A Hands-On Introduction to Hacking*. No Starch Press, 1 edition, 2014. ISBN 9781593275648.
- [62] John N. Whitley, Raphael C W Phan, Jie Wang, and David J. Parish. Attribution of attack trees. *Computers and Electrical Engineering*, 37(4):624–628, 2011. ISSN 00457906. doi: 10.1016/j.compeleceng.2011.04.010. URL <http://dx.doi.org/10.1016/j.compeleceng.2011.04.010>.

Appendix A

How to develop Maltego transforms

Maltego is a widely used program (*the program?*)¹ within the field of computer security for *Open Source Intelligence* (OSINT) gathering enabling an analyst to map technical infrastructure and information found across several sources (both the subject in question and third parties).

If you are reading this in order to start developing transforms for Maltego, I would recommend you to read this chapter in its entire length (and the links) first and then start over and follow along yourself, because while I tried to put it in a intuitive order (unlike Paterva did), you may get confused or have a lot of questions pop up along the way, which should be answered further down!

A.1 Maltego basics

If you read this thesis, there is a good chance you are acquainted with Maltego, but for the sake of completeness, here is a short run-down of Maltego basics:

Maltego is essentially a tool for passive reconnaissance in the context of computer security and social engineering, but are versatile enough to be used for whatever one might think of of other tasks within the domain.

There are several ways for an analyst to initiate research with Maltego. Most of these are automated in some way (with macros, called *machines*).

The basic, manual way is to open a new *graph*. This graph can be populated with one or more *entities* of some type (e.g. a domain or an IP), which then are filled out with some values (`domain.com` or `130.225.93.128`).

To each entity type, a number of *transforms* are provided by both Paterva and external developers; some of the externally developed transforms are paid, while others are free.

A transform performs some look-up on an OSINT-source or some computation on the entity and

¹There does not seem to be any alternatives: <https://alternativeto.net/software/maltego/> & <http://www.tpsort.com/similar-to/36881-top-15-maltego-alternative-and-similar-softwares>. Those listed here are either discontinued since long ago or have inferior capabilities. See also Sec. 2.1.2.2.

returns some information – usually in the form of one or more new entities of some type. The relation between the entity on which the transform were run and the returned output is marked by connecting lines, which the transform can weigh after importance (or the analyst can do that manually afterwards, supporting the concept of case-file management).

Different transforms can be run as many times as necessary on the same parent entity to gain further information. The researcher might also add findings from external tools (like the ones mentioned in Sec. 2.1.2.2) to Maltego to get the complete picture.

Further info on Maltego’s basic usage can be found in the Maltego user guide². In general, the guides are relatively up-to-date, but Maltego versions > 4.0 differs at a number of points and some info is duplicated; luckily Paterva are responsive on their support e-mail!

A.2 Making non-OEM transforms

Maltego comes equipped with a number of standard entities and transforms. It is however also possible for third party-providers to develop new entities and transforms for users. Depending on the requirements, this can be done strictly as local transforms or as public transforms, which are more versatile and easy to deploy across several systems³. It is possible to have your transforms displayed at the Transform Hub in Maltego if the transforms adhere to the advice/guidelines in [44] although it is not specified how to have them pre-distributed with the Maltego installation; you can always just issue a customer with the seed for your transforms.

For the purposes of this project, local transforms were not an option, so the following is written with respect to public transforms.

The setup and dataflow using public transforms can be seen in figure A.1 and is a bit funky; in essence Maltego queries an *internal Transform Distribution Server* (iTDS) (usually just Paterva’s own public), which in turn queries a transform host server hosted at the developer’s discretion. The Paterva iTDS⁴ only requires registration to access. The transform host server is setup in a Linux environment with Apache and Bottle (both webservers) and some example code⁵.

After setting up the servers, you can read the guides on writing and understanding transform coding. I suggest to read the more thorough one⁶ and then, if necessary, the one on the provided examples⁷.

To execute the transforms from the Maltego client, you need to connect the transform host

²<https://docs.paterva.com/en/user-guide/>

³See <https://docs.paterva.com/en/developer-portal/getting-started/> or <https://docs.paterva.com/en/developer-portal/tds-transforms/> for pro’s and con’s.

⁴Found at <https://cetas.paterva.com/TDS>

⁵See <https://docs.paterva.com/en/developer-portal/tds-transforms/transform-host-server-setup/> for step-by-step instructions and a examination of the supplied files in the TRX-files downloaded initially.

⁶<https://docs.paterva.com/en/developer-portal/tds-transforms/trx-transform-library-guide/>

⁷<https://docs.paterva.com/en/developer-portal/tds-transforms/tds-transforms-examples/tds-transform-example-code-python/>

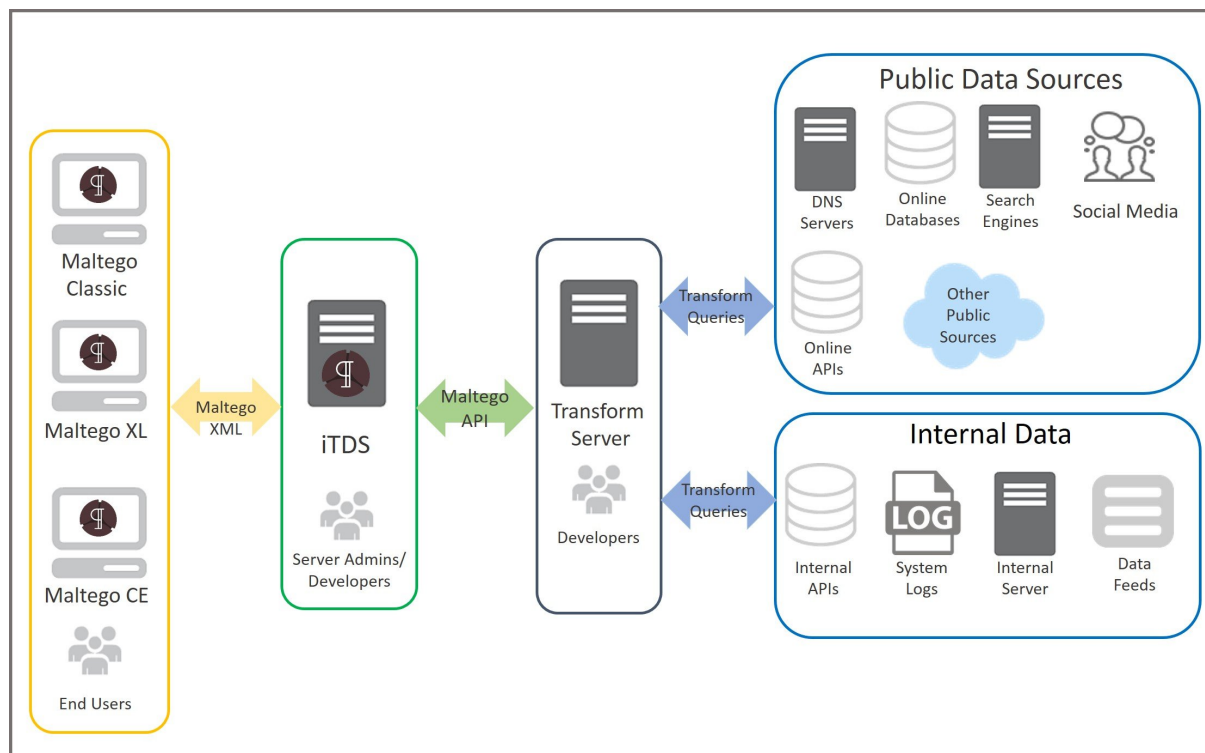


Figure A.1: The Maltego iTDS infrastructure and the dataflow in-between. The iTDS may be hosted locally, but this requires an authorized copy from Paterva. Image from Paterva developer portal.

server to the iTDS. This guide is hidden among the guides for local iTDS's here: <https://docs.paterva.com/en/server-guides/itds-server/itds-module/>. This guide also shows how to setup the Maltego client to use the developed third-party transforms.

When getting to the developing your own transforms, look to the TRX Library Reference⁸ for properties of the object parsed by the Maltego client (subsequently by the Paterva iTDS)⁹. Be aware that the XML-parser between the iTDS and your transform code (`Maltego.py`) is written after `python 2.x`-standards, so it uses `str()`, which only supports `ascii` (even though Maltego supports `UTF-8`); use the method `.encode("utf-8")` instead, which is the preferred way in `python 3.x`. By trial and error you should be able to substitute the method the appropriate places in the file if necessary.

Further guidance and advice on best practices can be found in [44] (a bit of mixed content on all of the above – useful to read before getting all caught up in writing transforms!).

A.2.1 Making custom entities

Custom entities are not coded as transforms are, but developed (according to Maltego's guidelines [44, 43]) in the Maltego client and exported as a configuration file. This file is then uploaded

⁸<https://docs.paterva.com/en/developer-portal/reference-guides/trx-library-reference/>

⁹I reported several spelling errors in this documentation. They will probably be corrected by the time of publishing this, but otherwise write the Paterva support!

and hosted on the transform host server

They are created by selecting *Entities* → *New Entity Type* in the ribbon. *Basic information* is filled out; it is important to choose *unique type name* to group custom entities [43]; this ensures the user can identify and delete them later (which can otherwise only be done by resetting the entire Maltego client). *Inheritance* is chosen as applicable; `maltego.Phrase` is used for many types of information requiring only some text to be saved, so it can be a good choice in many cases to enable many transforms (usually simple transforms searching somewhere) to be used on the custom entity.

Next, a custom property or the main property of the parent entity is chosen and finally it can be put into some category.

To add additional custom properties, select *Entities* → *Manage Entities* and select the custom entity in the list. Under the tab *Additional Properties* press *Add Property...* and add a *name* (to refer to in the transform code), a *display name* (shown in the *property view* when the entity is selected on a graph) and a data type.

Paterva have implemented default values for some properties based on other properties of the entity. Sometimes these will overwrite or hide whatever value the transform put into the property field directly. They suggested the following instead:

“[...] Unfortunately it does look like you can either set the name and city and country are blank OR you can set country and city and have the name automatically generated. Due to their defined default value, it isn't possible to remove the relationship between those properties.

I think the best solution would be to define a new property on your custom entity called 'Display'. You could then set the 'display' property as the display value on the graph, and set it's value to anything you want.

Then the 'name' property would automatically be set as <City>, <Country> but this value would no longer be displayed and could be ignored.”

–Mail-excerpt from Paterva Support July 2017

A.2.2 Distributing the transforms

Distribution can either happen via Paterva's transform host (requires the developer to buy a host) or as *seeds* (an URL) to be input manually in the Transform Hub in Maltego. The seed-URL is chosen on the iTDS and can be distributed as preferred by the developer.

The seed can both contain a configuration file (*Paired configuration* on the iTDS¹⁰) and transforms (and settings to present the user with as e.g. API-keys for transforms or the seed). The

¹⁰<https://docs.paterva.com/en/server-guides/itds-server/additional-functionality/#toc-paired-configuration>

configuration file is just an export from the Maltego client¹¹. It can contain both entities, transform sets, *machines* (macros), icons and API-keys for seeds. For simple transforms one typically only need to provide the custom entities used in the transforms of the seed (if any). For office environments a seed with a configuration file can be used to manage installations across several machines.

Update of transforms and other content via seeds in the Transform Hub happen automatically some unspecified time after changes has been published on the transform host server.

¹¹<https://docs.paterva.com/en/user-guide/ribbon-menu/entities-tab/#importing-and-exporting-entities>

Appendix B

Vulnerability reports

B.1 Qualys

This report is an (anonymous) output run on IP's of a client directly from the online Qualys vulnerability scanner and customized by the Dubex A/S Security Analytics Center¹ and sent as-is to the customers bundled with the executive (or “interpreted”) summary seen in Appendix B.2.

This is a full report in A4-format; it starts on the next page. . .

¹See <https://web.dubex.dk/services/security-analytics-center>

Technical Report (detailed)

file:///C:/Users/Rasmus/AppData/Local/Temp/maftemp-a25e9ef5/14...



Delivered by **Dubex:**

acme adhoc sårbarhedsscanning januar 2017

January 04, 2017

Rasmus Lau Petersen
 dubex-rl1
 Manager

Dubex A/S
 Gyngemose Parkvej 50
 Søborg, None DK-2860
 Denmark

01/01/0000 at 14:43:30 (GMT+0100)

Report Summary

Report Template: Technical Report (detailed)
 Sort by: Host
 IP Restriction: -
 Hosts Matching Filters: 4

References:

scan/1483498817.55401: 01/04/2017 at 04:01:08 (GMT+0100)

Summary of Vulnerabilities

Total: 60 Security Risk (Avg): 1.0

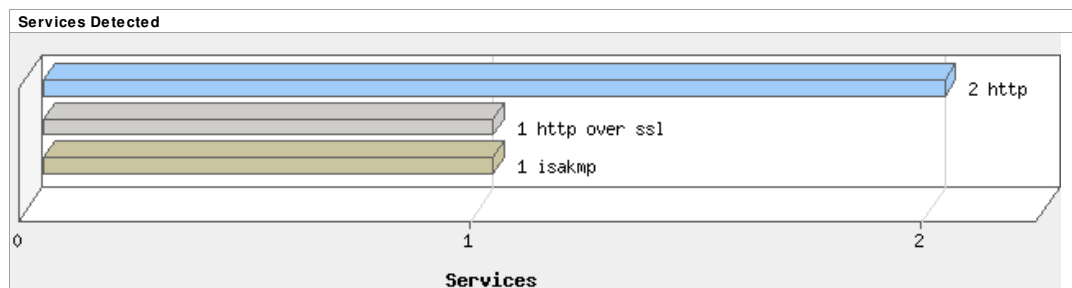
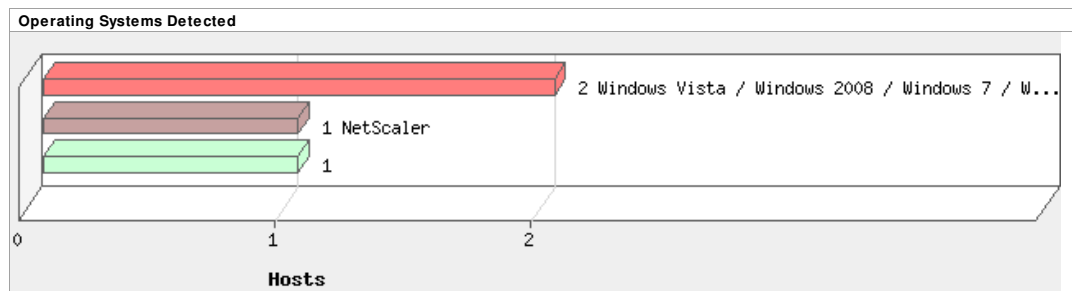
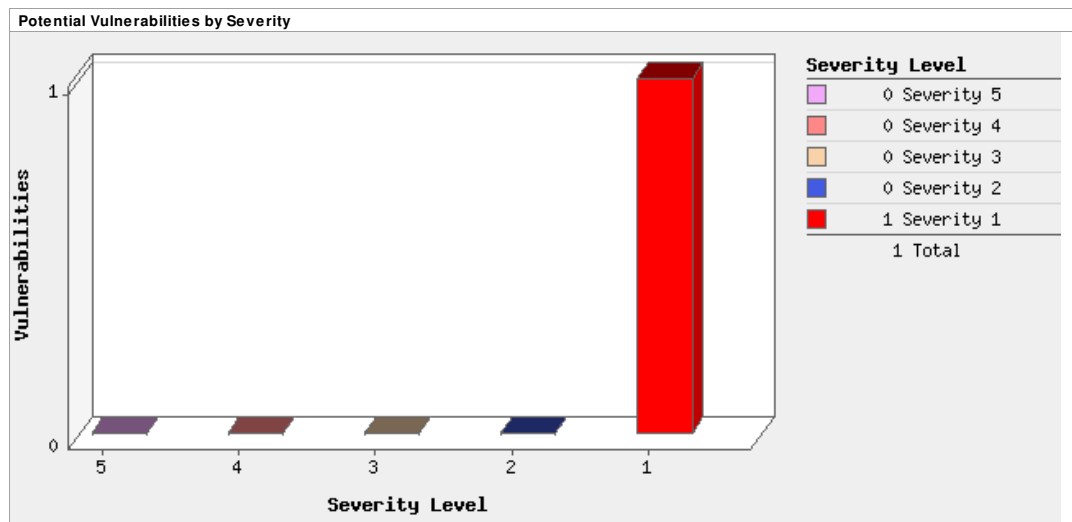
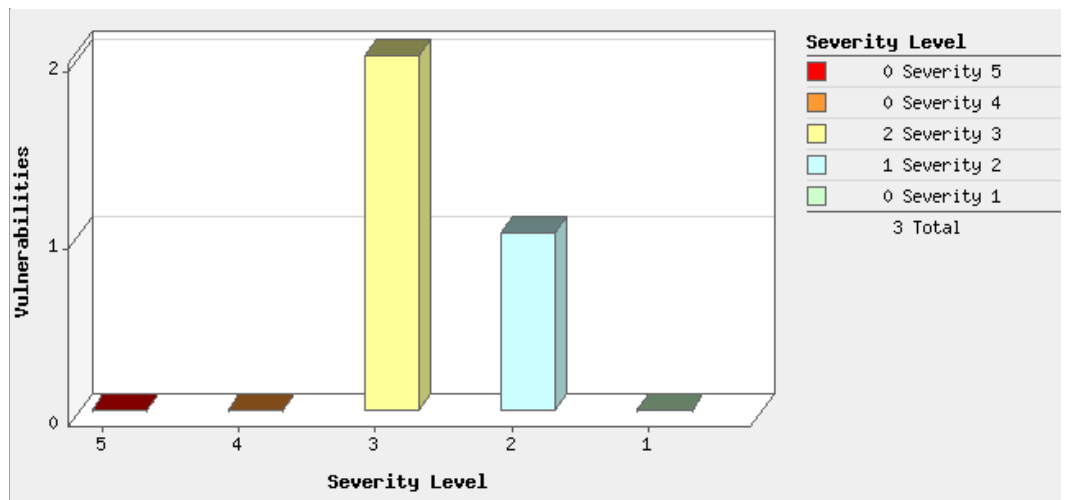
by Severity				
Severity	Confirmed	Potential	Information Gathered	Total
5	0	0	0	0
4	0	0	0	0
3	2	0	1	3
2	1	0	5	6
1	0	1	50	51
Total	3	1	56	60

5 Biggest Categories				
Category	Confirmed	Potential	Information Gathered	Total
Information gathering	0	0	23	23
TCP/IP	0	0	19	19
General remote services	3	1	6	10
Firewall	0	0	4	4
CGI	0	0	3	3
Total	3	1	55	59

Vulnerabilities by Severity

Technical Report (detailed)











file:///C:/Users/Rasmus/AppData/Local/Temp/maftemp-a25e9ef5/14...



Detailed Results

▼ 10.0.0.28 (-, -)

▼ Information Gathered (10)

- ▶  3 Remote Access or Management Service Detected
- ▶  1 DNS Host Name
- ▶  1 Firewall Detected
- ▶  1 Target Network Information
- ▶  1 Internet Service Provider
- ▶  1 Traceroute
- ▶  1 Host Scan Time
- ▶  1 Open UDP Services List
- ▶  1 ICMP Replies Received
- ▶  1 Host Name Not Available

▼ 10.0.0.125 (-, -)

NetScaler

▶ Information Gathered (12)

▼ 10.0.0.175 (-, -)

Windows Vista / Windows 2008 / Windows 7 / Windows 2012 / Windows 8 / Windows 10

▼ Vulnerabilities (3)

- ▼  3 SSL/TLS use of weak RC4 cipher port 443/tcp over SSL

QID: 38601
Category: General remote services
CVE ID: [CVE-2013-2566](#) [CVE-2015-2808](#)
Vendor Reference: -
Bugtraq ID: [91787](#), [58796](#)
Service Modified: 01/29/2016
User Modified: -
Edited: No
PCI Vuln: No
Ticket State:

THREAT:

Secure Sockets Layer (SSL v2/v3) and Transport Layer Security (TLS) protocols provide integrity, confidentiality and authenticity services to other protocols that lack these features. SSL/TLS protocols use ciphers such as AES,DES, 3DES and RC4 to encrypt the content of the higher layer protocols and thus provide the confidentiality service. Normally the output of an encryption process is a sequence of random looking bytes. It was known that RC4 output has some bias in the output. Recently a group of researchers has discovered that there is a stronger bias in RC4, which make statistical analysis of ciphertext more practical. The described attack is to inject a malicious javascript into the victim's browser that would ensure that there are multiple connections being established with a target website and the same HTTP cookie is sent multiple times to the website in encrypted form. This provides the attacker a large set of ciphertext samples, that can be used for statistical analysis. NOTE: On 3/12/15 NVD changed the CVSS v2 access complicity from high to medium. As a result Qualys revised the CVSS score to 4.3 immediately. On 5/4/15 Qualys is also revising the severity to level 3.

IMPACT:

If this attack is carried out and an HTTP cookie is recovered, then the attacker can use the cookie to impersonate the user whose cookie was recovered. This attack is not very practical as it requires the attacker to have access to millions of samples of ciphertext, but there are certain assumptions that an attacker can make to improve the chances of recovering the cleartext from ciphertext. For examples HTTP cookies are either base64 encoded or hex digits. This information can help the attacker in their efforts to recover the cookie.

SOLUTION:

RC4 should not be used where possible. One reason that RC4 was still being used was BEAST and Lucky13 attacks against CBC mode ciphers in SSL and TLS. However, TLSv 1.2 or later address these issues.

Technical Report (detailed)

file:///C:/Users/Rasmus/AppData/Local/Temp/maftemp-a25e9ef5/14...

COMPLIANCE:

Not Applicable

RESULTS:

CIPHER	KEY-EXCHANGE	AUTHENTICATION	MAC ENCRYPTION(KEY-STRENGTH)	GRADE
TLSv1 WITH RC4 CIPHERs IS SUPPORTED				
RC4-SHA	RSA	RSA	SHA1RC4(128)	MEDIUM
TLSv1.1 WITH RC4 CIPHERs IS SUPPORTED				
RC4-SHA	RSA	RSA	SHA1RC4(128)	MEDIUM
TLSv1.2 WITH RC4 CIPHERs IS SUPPORTED				
RC4-SHA	RSA	RSA	SHA1RC4(128)	MEDIUM

- ▶ 3 SSL/TLS Server supports TLSv1.0 port 443/tcp over SSL
- ▶ 2 SSL Certificate - Subject Common Name Does Not Match Server FQDN port 443/tcp over SSL

▶ **Information Gathered (20)**

▼ **10.0.0.230 (-, -)**

Windows Vista / Windows 2008 / Windows 7 / Windows 2012 / Windows 8 / Windows 10

▼ **Potential Vulnerabilities (1)**

▼ 1 Possible Scan Interference

QID: 42432
Category: General remote services
CVE ID: -
Vendor Reference: -
Bugtraq ID: -
Service Modified: 01/29/2016
User Modified: -
Edited: No
PCI Vuln: Yes
Ticket State:

THREAT:

Possible scan interference detected.
 A PCI scan must be allowed to perform scanning without interference from intrusion detection systems or intrusion prevention systems.
 The PCI ASV is required to post fail if scan interference is detected.
 The goal of this QID is to ensure that Active Protection Systems are not blocking, filtering, dropping or modifying network packets from a PCI Certified Scan, as such behavior could affect an ASV's ability to detect vulnerabilities. Active Protection Systems could include any of the following; IPS, WAF, Firewall, NGF, QoS Device, Spam Filter, etc. which are dynamically modifying their behavior based on info gathered from traffic patterns. This QID is triggered if a well known and popular service is not identified correctly due to possible scan interference. Services like FTP, SSH, Telnet, DNS, HTTP and Database services like MSSQL, Oracle, MySql are included.
 -If an Active Protection System is found to be preventing the scan from completing, Merchants should make the required changes (e.g. whitelist) so that the ASV scan can complete unimpeded.
 -If the scan was not actively blocked, Merchants can submit a PCI False Positive/Exception Request with a statement asserting that No Active Protection System is present or blocking the scan.
 Additionally, if there is no risk to the Cardholder Data Environment, such as no web service running, this can also be submitted as a PCI False Positive/Exception Request and reviewed per the standard PCI Workflow.
 For more details on scan interference during a PCI scan please refer to **ASV Scan Interference** section of [PCI DSS Approved Scanning Vendors Program Guide Version 2.0 May 2013 - page 14/28](#).

IMPACT:

If the scanner cannot detect vulnerabilities on Internet-facing systems because the scan is blocked by an IDS/IPS, those vulnerabilities will remain uncorrected and may be exploited if the IDS/IPS changes or fails.

SOLUTION:

Whitelist the Qualys scanner to scan without interference from the IDS or IPS.

Technical Report (detailed)

file:///C:/Users/Rasmus/AppData/Local/Temp/maftemp-a25e9ef5/14...

COMPLIANCE:

Not Applicable

RESULTS:

Service name: Unknown - Possible Scan Interference on TCP port 443.

► Information Gathered (14) 

▼ **Appendix**

Selected Scans

Scan

Launch Date: 01/04/2017 at 04:01:08 (GMT+0100)
 Active Hosts: 4
 Total Hosts: 6
 Type: Scheduled
 Status: Finished
 Reference: scan/1483498817.55401
 External Scanners: 64.39.102.197 (Scanner 9.0.29-1, Vulnerability Signatures 2.3.512-2)
 Duration: 00:14:18
 Title: acmeMonthly Vulnerability Test
 Asset Groups: acmeadhoc
 IPs: 10.0.0.28, 10.0.0.125, 10.0.0.143, 10.0.0.175, 10.0.0.181, 10.0.0.230
 Excluded IPs: -
 Option Profile: [Dubex SAC Profile](#)

Hosts Scanned

Successfully Scanned Hosts (IP)

10.0.0.28, 10.0.0.125, 10.0.0.175, 10.0.0.230

Target distribution across scanner appliances

External : 10.0.0.28, 10.0.0.125, 10.0.0.143, 10.0.0.175, 10.0.0.181, 10.0.0.230

Hosts Not Scanned

Hosts Not Alive (IP) (2)

10.0.0.143, 10.0.0.181

Options Profile

Dubex SAC Profile

Scan Settings

Ports	-
Scanned TCP Ports	Standard Scan
Scanned UDP Ports	Standard Scan
Scan Dead Hosts	On
Load Balancer Detection	Off
Perform 3-way Handshake	Off

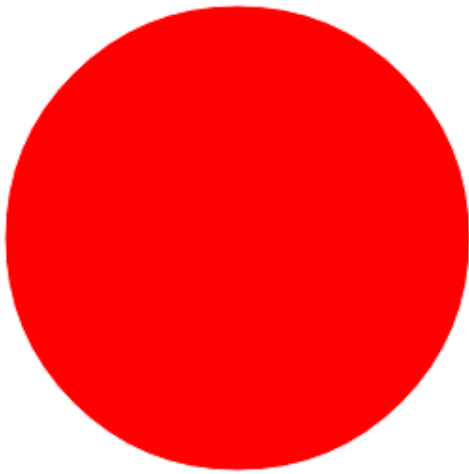




Technical Report (detailed) file:///C:/Users/Rasmus/AppData/Local/Temp/maftemp-a25e9ef5/14...

Vulnerability Detection	Complete
Password Brute Forcing	-
System	Standard
Custom	Disabled
Authentication	-
Windows	Disabled
Unix/Cisco	Disabled
Oracle	Disabled
Oracle Listener	Disabled
SNMP	Disabled
VMware	Disabled
DB2	Disabled
HTTP	Disabled
MySQL	Disabled
Overall Performance	Normal
Additional Certificate Detection	Normal
Authenticated Scan Certificate Discovery	Disabled
Hosts to Scan in Parallel	-
Use Appliance Parallel ML Scaling	Off
External Scanners	15
Scanner Appliances	30
Processes to Run in Parallel	-
Total Processes	10
HTTP Processes	10
Packet (Burst) Delay	Medium
Port Scanning and Host Discovery	-
Intensity	Normal
Dissolvable Agent	-
Dissolvable Agent (for this profile)	Disabled
Windows Share Enumeration	Disabled
Windows Directory Search	Disabled
Lite OS Discovery	Disabled
Advanced Settings	
Host Discovery	TCP Standard Scan and Additional TCP Ports: 1433, 1720, 3389, 5800, 5900, 3306, 10000 UDP Custom UDP Port List: 53, 123, 135, 137, 500, 1434 ICMP On
Packet Options	-
Ignore firewall-generated TCP RST packets	Off
Ignore all TCP RST packets	Off
Ignore firewall-generated TCP SYN-ACK packets	Off
Do not send TCP ACK or SYN-ACK packets during host discovery	Off

▶ **Report Filters**

▶ **Report Legend**

B.2 Dubex A/S executive summary

<p>Dubex: MANAGING RISK. ENABLING GROWTH. Klassifikation: Fortrolig mellem parterne</p> <p>Det Sikre Firma A/S</p> <p>Sårbarhedstest - december 2016</p> <div data-bbox="518 1243 1276 2004" style="text-align: center;"></div> <p>Dubex A/S : Gyngemose Parkvej 50 : 2860 Søborg : W www.dubex.dk : T +45 3283 0430 Side 1</p>	<p>Dubex: MANAGING RISK. ENABLING GROWTH.</p> <p>Testen</p> <p>Testen er udført 18. december 2016 mod 31 servere, hvoraf 28 svarede.</p> <p>Resultat af denne test</p> <p>I testen er der fundet:</p> <p>Sårbarheder i kategorier op til: "Højkritisk" </p> <p>Potentielle sårbarheder op til kategorien: "Kritisk" </p> <p>Muligheden for udnyttelse ligger højest i kategorien: "Nem" </p> <p>Samlet status er sat til "Rød" , da nogle af de alvorlige sårbarheder der er fundet kan være nemme at udnytte.</p> <p>Sårbarhederne og deres betydning er i dette dokument vurderet generelt. Risikoen skal endeligt vurderes i sammenhæng med servernes konkrete opsætning og anvendelse.</p> <p>En beskrivelse af de benyttede kategorier findes i bilaget sidst i rapporten.</p> <p>Dubex A/S : Gyngemose Parkvej 50 : 2860 Søborg : W www.dubex.dk : T +45 3283 0430 Side 2</p>
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------



Kommentarer til resultatet

Dette er den første test i en række af halvårige tests. Det fulde resultat for de forskellige testede hosts findes i den vedlagte detaljerede rapport.

De vigtigste sårbarheder som bør bearbejdes efter denne test er beskrevet nedenfor under "Specifikke resultater". Ved især de hosts der har kritiske eller højkritiske sårbarheder og mulighed for udnyttelse i kategorien "nem" eller "middel" bør man indkalkulere en risiko for at de allerede er blevet kompromitteret.

Det samlede resultat beskrives således:

- To hosts viser alvorlige sårbarheder vedrørende Remote Desktop Services.
- Der er på et par andre hosts risiko for misbrug af PhpMyAdmin og sårbarheder i PHP og Apache.
- Der findes en webhost med Heartbleed bug, som kan afsløre data fra tidligere brugeres SSL forbindelser til anonyme brugere fra Internettet.
- Der findes desuden et stort antal andre sårbarheder i forbindelse med SSL/TLS/kryptering.

Ved disse sårbarheder kan udnyttelse ikke ske uden særligt forarbejde, typisk ved at manipulere netværkstrafikken til at passere angriberen ("Man-in-the-middle" angreb). Selv om dette kan være vanskeligt er der tale om reelle sårbarheder. Det er muligt at fjerne dem med opdatering og/eller konfigurationsændringer.

- Resultatet fra bannercanning ("Information Gathered") viser at en host har en FTP server i en sårbar version (FileZilla Server version 0.9.41 beta).
- Nogle enheder afslører information, som ikke giver mulighed for at kompromittere dem, men som kan være nyttige for en angriber :
- Et par Cisco enheder afslører via NTP (Network Time Protocol) informationer, bl.a. om en intern IP adresse.
- Der er webservere der afslører deres private adresse i HTTP svar.
- Der er mulighed for at liste konfigurationsopslysninger for Front Page Extension.
- Der kan på nogle webservere tilsyneladende utilsigtet vises hvilke filer der findes i nogle mapper.

Shellshock

Testen har prøvet at udnytte den i september 2014 opdagede bash sårbarhed ("Shellshock", CVE-2014-6271) igennem velkendte "standard" cgi URL'er mod de testede webservere.

Der er ikke nogle positive svar på denne angrebsform, men der kan findes sårbare bash versioner på de testede Linux servere med mindre de er bash opdateret i det seneste kvartal.



Vurderet sårbarhedsniveau af hosts

Nedenstående er Dubex' vurdering af sårbarhedsniveauet på den enkelte host.

* = Se sektionen "Mulig scanningskonflikt" efter denne sektion.



Hosts hvor der findes sårbarheder eller risici der bør imødegås

95.x.x.19

95.x.x.26

95.x.x.27

95.x.x.29

95.x.x.58*

95.x.x.59*



Hosts hvor der findes sårbarheder der kan fjernes så sikkerheden forbedres

95.x.x.1

95.x.x.2

95.x.x.3

95.x.x.11

95.x.x.13

95.x.x.23


95.x.x.28

95.x.x.32

95.x.x.37

95.x.x.42

95.x.x.44
95.x.x.45
95.x.x.46
95.x.x.47
95.x.x.48
95.x.x.49
95.x.x.50
95.x.x.56
95.x.x.57
95.x.x.60
95.x.x.61



Hosts hvor der er ingen eller ubetydelige sårbarheder

95.x.x.10



Hosts der ikke svarede under testen

95.x.x.4
95.x.x.20
95.x.x.31

Mulig scanningskonflikt

På hosten 95.x.x.58 og 95.x.x.59 blev der registreret mulig indvirken på scanningsresultatet på portene 80/TCP og 443/TCP. Det anbefales at tillade Qualys-scanneren igennem et eventuelt IDS/IPS, da sårbarheder bag disse ellers ikke detekteres og kan blive angrebet, hvis IDS/IPS'en fejler.



Specifikke resultater

● **Sårbarhed: Windows RDP Remote Code Execution Vulnerability (MS12-020)** ████████

Risiko: Remote kodeafvikling eller denial-of-service gennem ondsindet trafik.
 Mulighed for udnyttelse: Medium ██████
 Rettelse: Opgrader softwaren, begræns netværksadgangen til RDP servicen.
 IP berørte: 95.x.x.58, 95.x.x.59

● **Sårbarhed: Windows RDP Web Access Elevation of Privilege Vulnerability (MS11-061)** ████████

Risiko: Cross-site-scripting gennem brugerens IE kan give afvikling af kommandoer på sitet.
 Mulighed for udnyttelse: Vanskelig ██████
 Rettelse: Opgrader softwaren, benyt XSS filter for relevant zone i IE.
 IP berørte: 95.x.x.58, 95.x.x.59

● **Sårbarhed: PHPMyAdmin Unauthorized Access Vulnerabilities** ████████

Risiko: Uautoriserede ændringer af databaser på serveren.
 Mulighed for udnyttelse: Nem ██████
 Rettelse: Opret adgangskontrol til PHPMyAdmin.
 IP berørt: 95.x.x.26 (port 80 og 443)

● **Flere sårbarheder: Outdated Software in Use** ██████

Risiko: Flere potentielle sårbarheder, der omhandler; utilsigtet afsløring af information, DoS og mulig kode afvikling.
 Udnyttelse: Medium ██████
 Rettelse: Opdater Apache og phpMyAdmin server software.
 IP berørt: 95.x.x.26



● **Sårbarhed: EOL/Obsolete Software: PHP 5.3.x** ████████

Risiko: Kendte sårbarheder rettes ikke da PHP 5.3.x har haft end-of-life
 Mulighed for udnyttelse: Medium ██████
 Rettelse: Opgradering til supporteret PHP version
 IP berørt: 95.x.x.29

● **Flere sårbarheder: Outdated Software in Use** ██████

Risiko: Flere potentielle sårbarheder, der omhandler utilsigtet afsløring af information, DoS og mulig kode afvikling.
 Udnyttelse: Medium ██████
 Rettelse: Opdater Apache server software.
 IP berørt: 95.x.x.29

● **Sårbarhed: OpenSSL Memory Leak Vulnerability (Heartbleed bug)** ████████

Risiko: Læsning udefra af OpenSSL memory med data fra tidligere transaktioner
 Mulighed for udnyttelse: Nem ██████
 Rettelse: Opgrader OpenSSL.
 IP berørt: 95.x.x.19 (port 13000)

● **Sårbarhed: OpenSSL Multiple Remote Security Vulnerabilities (TLS)** ████████

Risiko: Dekryptering af https datastrømmen af tredje part.
 Mulighed for udnyttelse: Meget vanskelig ██████
 Rettelse: Opgrader OpenSSL.
 IP berørte: 95.x.x.19 (port 13000), 95.x.x.26, 95.x.x.27

Den detaljerede Qualys rapport

Detaljerede oplysninger om testede enheder, med alle sårbarheder og resultater findes i den vedlagte Qualys rapporten, der indeholder 3 kategorier:

1. Confirmed (røde) - sikkerhedshuller som testen med sikkerhed har kunnet fastslå
2. Potential (gule) - sikkerhedshuller som kræver andre undersøgelser for at af- eller bekræfte om de findes
3. Information Gathered (blå) - oplysninger om versioner, filer osv. som er fundet under testen

Den detaljerede Qualys rapport kan være en "progress report", der gør det nemt at se forskellene mellem denne og en tidligere scanning af samme hosts, da der til højre for sårbarheden er en status for den givne sårbarhed som betyder:

"New" er en nyopdaget sårbarhed på systemet,
 "Active" er sårbarheder der stadig er tilstede og
 "Fixed" er sårbarheder der er blevet rettet.

Med venlig hilsen

Hassan Kallas
 Dubex A/S
 Tlf. +45 3283 0430
 www: <http://www.dubex.dk>

Bilag Beskrivelse af kategorier

Niveauer for sårbarheder - kort beskrevet

	Minimal	Angriberen kan samle oplysninger om hosten, informationen kan måske bruges til at finde andre sårbarheder.
	Medium	Angriberen kan samle løselomme oplysninger, f.eks. om sårbare softwareversioner.
	Alvorlig	Angriberen kan få adgang til information lagret på hosten og potentielt misbruge hosten.
	Kritisk	Angriberen kan få adgang til hosten eller løsløse information lagret på den.
	Højkritisk	Angriberen kan let få adgang til hosten hvilket kan føre til kompromittering af hele netværksikkerheden.

Røde ikoner benyttes for påviste sårbarheder
 Gule ikoner benyttes når der er risiko for en sårbarhed, men en undersøgelse er nødvendig for at fastslå om sårbarheden rent faktisk findes

Mulighed for udnyttelse

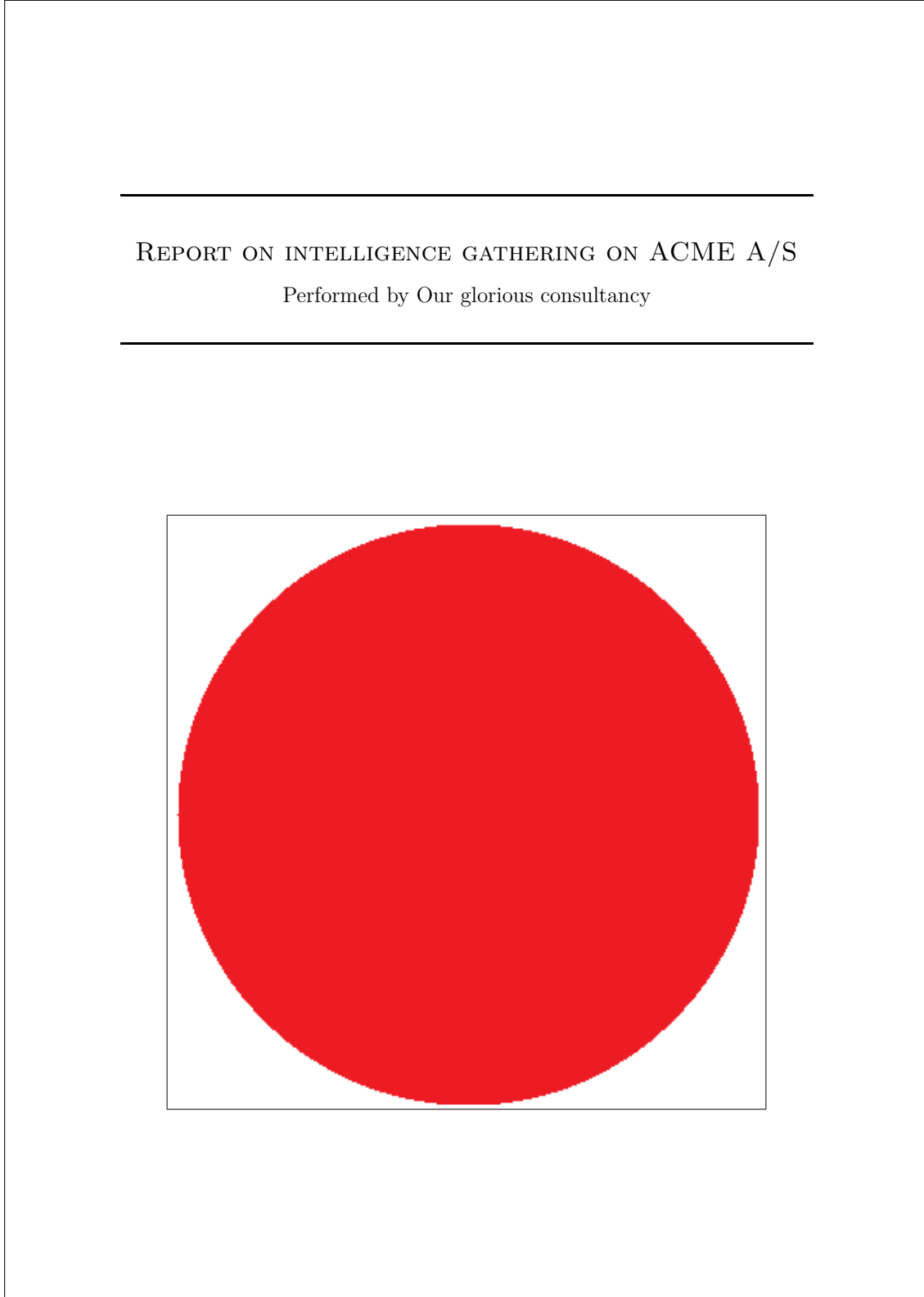
	Meget vanskelig	Angriberen skal have særlige forudsætninger og viden, f.eks. adgang til at manipulere netværkstrafik og viden om applikation
	Vanskelig	Angriberen skal have særlige forudsætninger, f.eks. mulighed for at manipulere netværkstrafik
	Medium	Angriberen skal have særlig adgang f.eks. mulighed for at aflytte trafik eller uprivilgeret adgang
	Nem	Angriberen kan umiddelbart benytte offentlig adgang til systemet

Samlet resultat

	Rød	Der findes kritiske sårbarheder som bør inoedegås
	Gul	Der findes sårbarheder som kan inoedegås, så sikkerheden forbedres
	Grøn	Der findes kun mindre sårbarheder

B.3 Example of an auto-generated report

Here follows an example of the auto-generated report made as part of this master thesis. This report is generated using the first 10 entries from the file depicted in Appendix D.2.



Intelligence gathering on ACME A/S

“Summer” 2017

1 Exective summary

Based on this report detailing the findings of an *Open Source Intelligence* gathering performed on ACME A/S, it is found that **ACME A/S is vulnerable to 4 of 5 common, OSINT-enabled cyber attack scenarios** reviewed and **violates 3 of 8 standards and guidelines**, which are expected to be applicable to ACME A/S as an organization operating in Denmark. This results in ● severity, which is not good!

10 findings from the OSINT-gathering were considered for this report.

The conclusions in this report is drawn from a number of commonly occurring scenarios and standards used and may not apply to ACME A/S directly. The results should be considered in a larger context with respect to the overall security maturity of ACME A/S and the risk appetite. Instead the results can be used to – in a simple way – understand the context in which the findings of the OSINT-gathering resides and enhance the understanding and procedures around OSINT-data and its influence on ACME A/S in daily business operations.

Contents

1	Executive summary	2
2	Introduction	4
3	Data found	4
3.1	Statistics on findings	4
3.1.1	Employee	5
3.1.2	SoMe	5
3.1.3	Non-personal internal	5
3.1.4	Supplier	5
3.1.5	Customer	5
4	Scenarios	7
4.1	Spear-phishing	7
4.1.1	Summary of findings	8
4.1.2	Individual requirements	8
4.2	In-person attacks	9
4.2.1	Summary of findings	9
4.2.2	Individual requirements	10
4.3	CEO-fraud	10
4.3.1	Summary of findings	11
4.3.2	Individual requirements	11
4.4	Subverting the supply chain	11
4.4.1	Summary of findings	12
4.4.2	Individual requirements	12
4.5	Targeted (D)DoS	12
4.5.1	Summary of findings	13
4.5.2	Individual requirements	13
5	Standards	14
5.1	Federal CIO Council	14
5.1.1	Individual controls/policies/rules	15
5.2	DS/ISO 27001 – Direct violations	15
5.2.1	Individual controls/policies/rules	16
5.3	Mitnick’s guidelines	16
5.3.1	Individual controls/policies/rules	16

2 Introduction

This report is auto-generated from the findings (data) of a Maltego-investigation performed by Our glorious consultancy towards the company ACME A/S.

The findings come from a gathering of *open source intelligence* (OSINT). OSINT is *all* publicly available information found across many freely available sources – it may be *footprints* of the organization and its employee's daily operations (e.g. from public registers (government or 3rd party)), a product of use of IT systems, web content (e.g. articles, documents and their meta-data), news or active information sharing by individual employees on e.g. social media and fora. Some of the data are avoidable, some are not, but their value to an attack cannot be known until it enters a greater context of an attacker's knowledge and intentions.

To find the information, the attacker can use search engines like Google and Shodan, but also the organizations' own sites, government sites or public registries. The information found is then utilized to try to exploit human psychological mechanisms (i.e. "social engineering") to e.g. establish context with an employee s.t. they place an unmerited degree of trust on an object/subject (e.g. a received e-mail or a person addressing them).

The report suggests how the data found relates to a range of common, targeted cyber attack scenarios enabled by OSINT-data as well as applicable guidelines to organizations acting under Danish legislation.

The scenarios and guidelines are chosen based on the analysis made in master thesis on the subject on DTU Compute summer 2017.

The report is organized into three parts:

- Section 3 categorizes the input-data into 5 different primary categories of information. In each subsection, the sublabels per primary category are listed as well as the count of the findings categorized under each sublabel.
- Section 4 lists 5 common OSINT-enabled cyber attack scenarios which the input-data are considered against. Each scenario is put into a real-world context with an explanation of the scenario and which OSINT-data can go into enabling an attacker to exploit it. For each scenario, it can be seen if ACME A/S are presumed to be vulnerable to the scenarios based on the findings. Additionally we list the input-data, which were found to be contributing to the specific requirements deemed to enable such an attack.
- Section 5 lists 8 standards and guidelines, which are expected to be applicable to the operations of ACME A/S as a Danish organization. For each standard/guideline the policies/controls pertaining to findings such as those appearing here, are listed. If these are violated based on the findings, this is shown with the findings violating.

3 Data found

This section lists the data input to this report. The data is grouped into five categories, each having a number of subcategories which is used to recognize the data in relation to the common cyber attack scenarios and the guidelines covered by this report.

3.1 Statistics on findings

In the tables following, statistics of the findings made during the investigation is given. A bar graph giving an overview of the data within the 5 categories of information, is found in Figure 1.

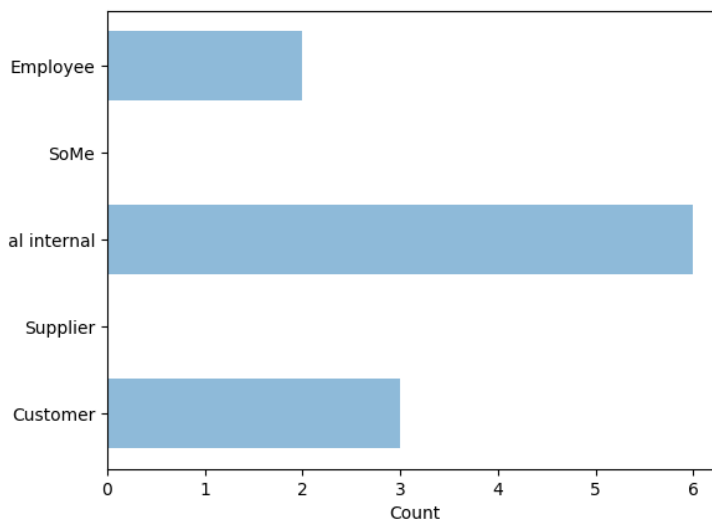


Figure 1: The distribution of findings within the 5 categories of information used in this report.

3.1.1 Employee

Data label	Count
Name/job positions	1
Personal information :: Other private associations (school, hobby, sport, polical)	1

Table 1: The count of all findings within the category Employee

3.1.2 SoMe

There are no findings within this category.

3.1.3 Non-personal internal

Data label	Count
Physical/virtual infrastructure :: Type/versions	1
Physical/virtual infrastructure :: Names (DNS, nicknames)	3
Physical/virtual infrastructure :: Manufacturer/provider	1
Physical/virtual infrastructure :: IP's :: External IP	1

Table 2: The count of all findings within the category Non-personal internal

3.1.4 Supplier

There are no findings within this category.

3.1.5 Customer

Intelligence gathering on ACME A/S

“Summer” 2017

Data label	Count
Customer names	2
Other customer information	1

Table 3: The count of all findings within the category Customer

4 Scenarios

This section relates the findings of Section 3 to 5 common OSINT-enabled cyber attack scenarios.

The scenarios are created as a part of a master thesis project. They seek to cover a wide variety of OSINT-enabled attacks, but is important to understand that it is impossible to describe *all* attack scenarios enabled by the findings used as input to this auto-generated report. These attacks (and the persons behind) employ a vast range of knowledge and information; if some information were not acquired during the research, a specific variety of a scenario could go completely overlooked.

The findings in this section should instead be used as guideline to understand which circumstances or specific data can enable a OSINT-enabled attack against ACME A/S.

5 common cyber attack scenarios were considered. ACME A/S are considered vulnerable to them as shown in Table 4. Details of each of the 5 scenarios and the findings that lead to this conclusion, are found in the following subsections considering each scenario individually.

Cyber attack scenario	Vulnerable?
Spear-phishing	True
In-person attacks	True
CEO-fraud	False
Subverting the supply chain	True
Targeted (D)DoS	True

Table 4: The cyber attack scenarios considered in this report and whether ACME A/S are considered vulnerable towards each of them.

4.1 Spear-phishing

Carried out mostly through emails as the easiest attack vector, but also phone calls, face-to-face or through other means of communication (as people may recognize voice or biometrics of the impersonated person/organization); also called *pretexting*.

The goal is information disclosure for further attacks, directly for e.g. monetary gain (through encouraging bank transfers, acquiring passwords, (bank) account information or NemID-keys) or delivery of a attack payload for e.g. espionage or activism or any other goal.

The most important differences from un-targeted phishing attacks, is that they target a few, specific receivers, put more work into creating a credible email/relation through language, logo's, current activities/contacts of the organization and non-threatening content. However, while they may seek to imitate language of e.g. a professional email or invoice, another trait used in the emails is a sense of urgency and/or secrecy to convince the receiver to perform the task fast (e.g. a bank transfer) and without disclosing anything to colleagues.

To improve credibility, the attacker can employ OSINT to discover:

- Current professional relations (e.g. suppliers, collaborators or customers) found on e.g. LinkedIn, Facebook, public forums, job advertisements (describing technical qualifications needed of new hires) or homepage of the organization or their vendors/customers.
- Private relations or economic interests found on the aforementioned sources or through e.g. public leak data including company domain email addresses.
- Employee names and private e-mail addresses (from e.g. social media accounts) to deliver a malicious payload circumventing organizational countermeasures.
- Specifics of the organization's structure from e.g. informative organizational chart, job postings, points of contact (for homepage, support or legals) or meta-data from documents

Intelligence gathering on ACME A/S

“Summer” 2017

on the organization’s homepage. Specifics can include names, positions, job titles, phone numbers, location (e.g. for using the target’s national language or TLD) etc. Phone numbers can also serve as an alternative contact medium, where the attacker will then employ other parts of the collected OSINT.

- Knowledge of organizational operations (in addition to the previously mentioned) like travel plans, current issues (from public forums or bug reports).

The attacker can of course also employ technical solutions to increase credibility by e.g. acquiring access to email servers. This requires prior use of social engineering to gather passwords, deliver malicious payloads or similar.

Examples of attack are invoice fraud with fake invoices looking to come from real vendors, coaxing employees into sending money to “colleagues” or trying to get further information on the organization/employees. The most repeated advice in government guidance to hinder these kinds of attacks, are to implement specific procedures of double-checking e.g. money transfers and information disclosures by calling the responsible or the sender and general vigilance of employees.

4.1.1 Summary of findings

Overall it is found that ACME A/S is vulnerable to this common social engineering attack scenario.

This scenario consists of 6 requirements of which it is expected that 3 of the requirements need to be satisfied in order for ACME A/S to be vulnerable.

During the investigation on ACME A/S, data satisfying 4 of the scenarios’ requirements were identified.

In Table 5 each requirement for the scenario “Spear-phishing” is listed. In the following subsections, all data found to be contributing to satisfying the specific attack scenario requirement are listed, s.t. it is possible to gain an insight into exactly what piece of OSINT-data contributed to these findings.

Requirements for this scenario	Satisfied?
Employee names/position	True
Organizational structure	True
Supplier/customer or other professional relations	True
Personal/corporate email addresses, phone numbers	False
Relations (friends, hobbies)	True
Typosquatting domains	False

Table 5: The individual requirements for the cyber attack scenario “Spear-phishing” and the findings of the intelligence gathering satisfying them.

The individual requirements satisfied are considered so by the following findings:

4.1.2 Individual requirements

- ‘Employee names/position’ is considered satisfied due the following findings:
 - ‘Lars Estes Henriksen’ (*found from/in/on “ESTES”*)
- ‘Organizational structure’ is considered satisfied due the following findings:
 - ‘Lars Estes Henriksen’ (*found from/in/on “ESTES”*)
- ‘Supplier/customer or other professional relations’ is considered satisfied due the following findings:

Intelligence gathering on ACME A/S

“Summer” 2017

- ‘djoef-dk.mx1.comendosystems.com’ (*found from/in/on “djoef.dk”*)
- ‘Personal/corporate email addresses, phone numbers’ is not considered satisfied based on the findings.
- ‘Relations (friends, hobbies)’ is considered satisfied due the following findings:
 - ‘Nissan X-Trail DIG-T 163 SUV 2WD 6 M/T’ (*found from/in/on “ESTES”*)
- ‘Typosquatting domains’ is not considered satisfied based on the findings.

4.2 In-person attacks

If the attacker is willing to interact directly with the target/human sources of information in general by e.g. appearing physically on location or calling, an even wider range of scenarios are possible. These are naturally targeted in nature, as the attacker must choose some specific organization/place to appear physically.

Most scenarios described built upon spear-phishing attacks, but requires human interaction, methodically planning and agility of the attack plan. The attacks are diverse in their necessity of information required to work, but all exploit the human mind (i.e. *social engineering*) by different methods.

An example describes how the attacker through human interaction by phone only acquires internal hostnames, credentials to these, out-of-office voice-mails, phone (with internal extensions) and fax numbers, VPN-access and in the end, the data of some project. For this attack, the information found from OSINT-sources beforehand was only:

- Some personal data to verify with (date of birth, family info, social security number etc.).
- Employees of different departments (only a few were necessary, the rest can be offered by the employees called by *namedropping*).
- Company locations/sites.

The rest of the information were discovered during the course of the attack. It should however be noted, that this story involves violation of many policies implemented in modern organization with controls such as those in DS/ISO 27001; those might however fail if the awareness among the employees are not sufficient.

Another method could be for the attacker to show up on premises, which requires proper attire of employees, vendors, shipping handlers etc., and maybe some knowledge of company behavior or locations; afterwards he can use social engineering-techniques to recover the necessary information.

Attacks of this type not requiring any particular OSINT-data includes baiting with infectious USB-devices dropped on the organization’s parking/grounds, tailgating (following employees) inside the organization’s premises or *dumpster diving* to recover confidential information.

4.2.1 Summary of findings

Overall it is found that ACME A/S is vulnerable to this common social engineering attack scenario.

This scenario consists of 5 requirements of which it is expected that 2 of the requirements need to be satisfied in order for ACME A/S to be vulnerable.

During the investigation on ACME A/S, data satisfying 3 of the scenarios’ requirements were identified.

In Table 6 each requirement for the scenario “In-person attacks” is listed. In the following subsections, all data found to be contributing to satisfying the specific attack scenario requirement

Intelligence gathering on ACME A/S

"Summer" 2017

Requirements for this scenario	Satisfied?
Other personal identifying information	True
Other employee names (name dropping)	True
Internal contact information	False
Internal infrastructure ((host-)names, type, IP's etc.)	True
Supplier information (names, relation to organization)	False

Table 6: The individual requirements for the cyber attack scenario "In-person attacks" and the findings of the intelligence gathering satisfying them.

are listed, s.t. it is possible to gain an insight into exactly what piece of OSINT-data contributed to these findings.

The individual requirements satisfied are considered so by the following findings:

4.2.2 Individual requirements

- **'Other personal identifying information'** is considered satisfied due the following findings:
 - 'Nissan X-Trail DIG-T 163 SUV 2WD 6 M/T' (*found from/in/on "ESTES"*)
- **'Other employee names (name dropping)'** is considered satisfied due the following findings:
 - 'Lars Estes Henriksen' (*found from/in/on "ESTES"*)
- **'Internal contact information'** is not considered satisfied based on the findings.
- **'Internal infrastructure ((host-)names, type, IP's etc.)'** is considered satisfied due the following findings:
 - '89.104.206.4' (*found from/in/on "djoef-dk.mx1.comendosystems.com"*)
 - 'djoef-dk.mx1.comendosystems.com' (*found from/in/on "djoef.dk"*)
 - 'edit.djoef.dk' (*found from/in/on "djoef.dk"*)
 - 'https://www.djoef.dk/ /media/documents/djoef/f/forside.ashx?la=da www.djoef.dk' (*found from/in/on "djoef.dk"*)
- **'Supplier information (names, relation to organization)'** is not considered satisfied based on the findings.

4.3 CEO-fraud

Considered a specific kind of spear-phishing, this attack impersonates or targets C-level employees – also called *whales*, as they are "the big targets".

The aim is to perform acts similar to spear-phishing, but due to the large amount of money that may be involved with C-level roles, a larger reward can be collected by the attacker.

Prerequisites and traits of the attack are similar to spear-phishing as well; it may however *not be necessary to know any vendors/customers of the organization* for this attack, but only:

- Name of head of the company.
- His e-mail address (to mimic or create something similar).
- Managers/employees authorized to perform a transfer of funds.

Intelligence gathering on ACME A/S

"Summer" 2017

The emails may be even better crafted than regular spear-phishing emails through e.g. more formal/correct language.

Examples of *spear-phishing* can also be considered whaling; in a specific example, The National Museum of Art in Denmark were recently phished for 805.000 DKK by impersonating the CEO and targeting an employee with privileged access to the accounts.

4.3.1 Summary of findings

Overall it is found that ACME A/S is not vulnerable to this common social engineering attack scenario.

This scenario consists of 3 requirements of which it is expected that 2 of the requirements need to be satisfied in order for ACME A/S to be vulnerable.

During the investigation on ACME A/S, data satisfying 1 of the scenarios' requirements were identified.

In Table 7 each requirement for the scenario "CEO-fraud" is listed. In the following subsections, all data found to be contributing to satisfying the specific attack scenario requirement are listed, s.t. it is possible to gain an insight into exactly what piece of OSINT-data contributed to these findings.

Requirements for this scenario	Satisfied?
Name/e-mail of CEO/CFO	False
Name/e-mail on (privileged) employee	False
Name/e-mail of some partner	True

Table 7: The individual requirements for the cyber attack scenario "CEO-fraud" and the findings of the intelligence gathering satisfying them.

The individual requirements satisfied are considered so by the following findings:

4.3.2 Individual requirements

- 'Name/e-mail of CEO/CFO' is not considered satisfied based on the findings.
- 'Name/e-mail on (privileged) employee' is not considered satisfied based on the findings.
- 'Name/e-mail of some partner' is considered satisfied due the following findings:
 - 'djoef-dk.mx1.comendosystems.com' (*found from/in/on "djoef.dk"*)

4.4 Subverting the supply chain

This is to attack equipment or software being delivered to the organization. The goal is to deliver a payload through the regular supply chain of the organization; suppliers which the organization has already put a high level of trust in and perhaps thus are less likely to question deliveries/content from.

We know NSA performs this practice against hardware/servers exported from the US and some believe that Huawei-equipment does the same. In a specific attack, a vendor of scanners running Microsoft XP Embedded OS were shipped with malware. The malware targeted ERP-systems of shipping and logistics and later also manufacturers.

A lot of information may have gone into compromising the manufacturer himself, but from the actual target organization, it may only necessary to know:

- A type of software used in the company (e.g. an ERP-system).

Intelligence gathering on ACME A/S

"Summer" 2017

- A type of hardware deployed or the distributor bought from.
- The attack could also be leveraged by identifying the employee responsible of procurement of IT equipment (and contact information) e.g. from an organizational chart of social media.

4.4.1 Summary of findings

Overall it is found that ACME A/S is vulnerable to this common social engineering attack scenario.

This scenario consists of 3 requirements of which it is expected that 2 of the requirements need to be satisfied in order for ACME A/S to be vulnerable.

During the investigation on ACME A/S, data satisfying 2 of the scenarios' requirements were identified.

In Table 8 each requirement for the scenario "Subverting the supply chain" is listed. In the following subsections, all data found to be contributing to satisfying the specific attack scenario requirement are listed, s.t. it is possible to gain an insight into exactly what piece of OSINT-data contributed to these findings.

Requirements for this scenario	Satisfied?
Employee responsible of procurement (or other specific department)	False
Some software used in the organization (and the supplier)	True
Some hardware used in the organization (and the supplier)	True

Table 8: The individual requirements for the cyber attack scenario "Subverting the supply chain" and the findings of the intelligence gathering satisfying them.

The individual requirements satisfied are considered so by the following findings:

4.4.2 Individual requirements

- **'Employee responsible of procurement (or other specific department)'** is not considered satisfied based on the findings.
- **'Some software used in the organization (and the supplier)'** is considered satisfied due the following findings:
 - 'djoef-dk.mx1.comendosystems.com' (*found from/in/on "djoef.dk"*)
 - 'edit.djoef.dk' (*found from/in/on "djoef.dk"*)
 - 'https://www.djoef.dk/ /media/documents/djoef/f/forside.ashx?la=da www.djoef.dk' (*found from/in/on "djoef.dk"*)
- **'Some hardware used in the organization (and the supplier)'** is considered satisfied due the following findings:
 - 'djoef-dk.mx1.comendosystems.com' (*found from/in/on "djoef.dk"*)

4.5 Targeted (D)DoS

This attack can be employed if the attacker is politically motivated and wants to shut down a service/website/operations of the organization, but also as a tool of extortion. The danger of this is inherent in servers connected to the Internet; hostnames are quick to resolve and target, but if public IP's not meant to be exposed/used by regular users, are found through e.g. Shodan or public pastes of stolen data, the right security measures might not be present.

Intelligence gathering on ACME A/S

“Summer” 2017

It can also be a problem if internal IP-addresses are found from e.g. internal documents or in website descriptions, as it can be used to claim credibility (by proving knowledge of internal network components). Additionally, techniques exist to route traffic through public IP’s to unintended servers inside the network.

4.5.1 Summary of findings

Overall it is found that ACME A/S is vulnerable to this common social engineering attack scenario.

This scenario consists of 2 requirements of which it is expected that 1 of the requirements need to be satisfied in order for ACME A/S to be vulnerable.

During the investigation on ACME A/S, data satisfying 1 of the scenarios’ requirements were identified.

In Table 9 each requirement for the scenario “Targeted (D)DoS” is listed. In the following subsections, all data found to be contributing to satisfying the specific attack scenario requirement are listed, s.t. it is possible to gain an insight into exactly what piece of OSINT-data contributed to these findings.

Requirements for this scenario	Satisfied?
Internal IP’s or hostnames	False
External IP’s	True

Table 9: The individual requirements for the cyber attack scenario “Targeted (D)DoS” and the findings of the intelligence gathering satisfying them.

The individual requirements satisfied are considered so by the following findings:

4.5.2 Individual requirements

- ‘**Internal IP’s or hostnames**’ is not considered satisfied based on the findings.
- ‘**External IP’s**’ is considered satisfied due the following findings:
 - ‘89.104.206.4’ (*found from/in/on “djoef-dk.mx1.comendosystems.com”*)

Intelligence gathering on ACME A/S

"Summer" 2017

5 Standards

This section relates the findings of Section 3 to 8 standards and guidance applicable to ACME A/S as an organization operating under Danish legislation.

The standards and guidance are identified through a master thesis project, where industry standards and -guidelines from renowned institutions were considered. In particular, material applicable to Danish organizations were considered.

The material chosen are published by government bodies as NCSC (UK), Federal CIO Council (US) and CFCS (DK) and standardization groups as NIST (US) and the ISO-group. Finally, industry veteran Kevin Mitnick's guidance are used.

The content of this section aim to *give guidance* to ACME A/S to understand the findings in context of the standards and guidance that are likely to influence the daily operations of the organization as imposed by legislation or in other ways.

The stanarads and guidance in turn can help shed a different light on how specific data (under some circumstances) or can enable a OSINT-enabled attack against ACME A/S.

8 standards/guidance were considered – each consisting of a number of controls/policies/rules. ACME A/S are considered vulnerable to a standard/guideline of one controls/policies/rules are violated. We consider ACME A/S in violation as shown in Table 10.

Details of each of the 8 standards/guidance and the findings that lead to this conclusion, are found in the following subsections considereing each standard individually¹

Standard/guideline	Violated?
NIST	False
NCSC	False
CFCS	False
CPNI	False
Agency for Digitisation	False
Federal CIO Council	True
DS/ISO 27001 – Direct violations	True
Mitnick's guidelines	True

Table 10: The standards/guidance considered in this report and whether ACME A/S are considered in violation of them.

5.1 Federal CIO Council

Overall it is found that ACME A/S is in violation of this standard/guideline.

This standard/guideline consists of 2 individual controls/policies/rules; if one of these are violated, we consider ACME A/S in violation of this particular standard/guideline.

During the investigation on ACME A/S, data satisfying 2 of the controls/policies/rules were identified.

In Table 11 each control/policy/rule for the standard/guideline "Federal CIO Council" is listed. In the following subsections, all data found to be contributing to violating the specific control/policy/rule are listed, s.t. it is possible to gain an insight into exactly what piece of OSINT-data contributed to these findings.

¹Please note that standard/guideline 1-5 are not reflected in the current implementation of the code generating

Intelligence gathering on ACME A/S

“Summer” 2017

Control/policy/rule within this standard/guideline	Violated?
Separate professional and personal life: Don't use corp. email addresses for personal accounts (SoMe, school contact sheet etc.) – especially not in relation with other personal details	True
Present yourself properly online (including not disclosing valuable information to an adversary)	True

Table 11: The controls/policies/rules of the standard/guideline “Federal CIO Council” and the findings of the intelligence gathering considered in violation of them.

The individual controls/policies/rules are considered violated by the following findings:

5.1.1 Individual controls/policies/rules

- The control/policy/rule ‘**Separate professional and personal life: Don't use corp. email addresses for personal accounts (SoMe, school contact sheet etc.) – especially not in relation with other personal details**’ is considered violated based on the following findings:
 - ‘Nissan X-Trail DIG-T 163 SUV 2WD 6 M/T’ (*found from/in/on “ESTES”*)
- The control/policy/rule ‘**Present yourself properly online (including not disclosing valuable information to an adversary)**’ is considered violated based on the following findings:
 - ‘Lars Estes Henriksen’ (*found from/in/on “ESTES”*)

5.2 DS/ISO 27001 – Direct violations

Overall it is found that ACME A/S is in violation of this standard/guideline.

This standard/guideline consists of 5 individual controls/policies/rules; if one of these are violated, we consider ACME A/S in violation of this particular standard/guideline. During the investigation on ACME A/S, data satisfying 3 of the controls/policies/rules were identified.

In Table 12 each control/policy/rule for the standard/guideline “DS/ISO 27001 – Direct violations” is listed. In the following subsections, all data found to be contributing to violating the specific control/policy/rule are listed, s.t. it is possible to gain an insight into exactly what piece of OSINT-data contributed to these findings.

Control/policy/rule within this standard/guideline	Violated?
Protect personal identifying information	True
Records shall be protected from unauthorized access	False
Transfer policies/controls for data shall be in place	False
Procedures for handling classified information shall be implemented (i.e. classified information should not be publicly available)	True
Information involved in electronic messaging shall be appropriately protected	True

Table 12: The controls/policies/rules of the standard/guideline “DS/ISO 27001 – Direct violations” and the findings of the intelligence gathering considered in violation of them.

The individual controls/policies/rules are considered violated by the following findings:

this report.

Intelligence gathering on ACME A/S

“Summer” 2017

5.2.1 Individual controls/policies/rules

- The control/policy/rule ‘**Protect personal identifying information**’ is considered violated based on the following findings:
 - ‘Renault Captur dCi 90’ (*found from/in/on “AB12345”*)
 - ‘Sabri Elhaj Moussa’ (*found from/in/on “AB12345”*)
 - ‘Catharina Estes Henriksen’ (*found from/in/on “ESTES”*)
 - ‘Lars Estes Henriksen’ (*found from/in/on “ESTES”*)
 - ‘Nissan X-Trail DIG-T 163 SUV 2WD 6 M/T’ (*found from/in/on “ESTES”*)
- ‘**Records shall be protected from unauthorized access**’ is not considered violated based on the findings.
- ‘**Transfer policies/controls for data shall be in place**’ is not considered violated based on the findings.
- The control/policy/rule ‘**Procedures for handling classified information shall be implemented (i.e. classified information should not be publicly available)**’ is considered violated based on the following findings:
 - ‘Renault Captur dCi 90’ (*found from/in/on “AB12345”*)
 - ‘Sabri Elhaj Moussa’ (*found from/in/on “AB12345”*)
 - ‘Catharina Estes Henriksen’ (*found from/in/on “ESTES”*)
- The control/policy/rule ‘**Information invovled in electronic messaging shall be appropriately protected**’ is considered violated based on the following findings:
 - ‘djoef-dk.mx1.comendosystems.com’ (*found from/in/on “djoef.dk”*)
 - ‘edit.djoef.dk’ (*found from/in/on “djoef.dk”*)
 - ‘https://www.djoef.dk/ /media/documents/djoef/f/forside.ashx?la=da www.djoef.dk’ (*found from/in/on “djoef.dk”*)

5.3 Mitnick’s guidelines

Overall it is found that ACME A/S is in violation of this standard/guideline.

This standard/guideline consists of 5 individual controls/policies/rules; if one of these are violated, we consider ACME A/S in violation of this particular standard/guideline.

During the investigation on ACME A/S, data satisfying 3 of the controls/policies/rules were identified.

In Table 13 each control/policy/rule for the standard/guideline “Mitnick’s guidelines” is listed. In the following subsections, all data found to be contributing to violating the specific control/policy/rule are listed, s.t. it is possible to gain an insight into exactly what piece of OSINT-data contributed to these findings.

The individual controls/policies/rules are considered violated by the following findings:

5.3.1 Individual controls/policies/rules

- ‘**No organizational details on 3rd party sites (of policies, infrastructure, contact information)**’ is not considered violated based on the findings.
- The control/policy/rule ‘**No info on organizational structure or job positions**’ is considered violated based on the following findings:

Intelligence gathering on ACME A/S

“Summer” 2017

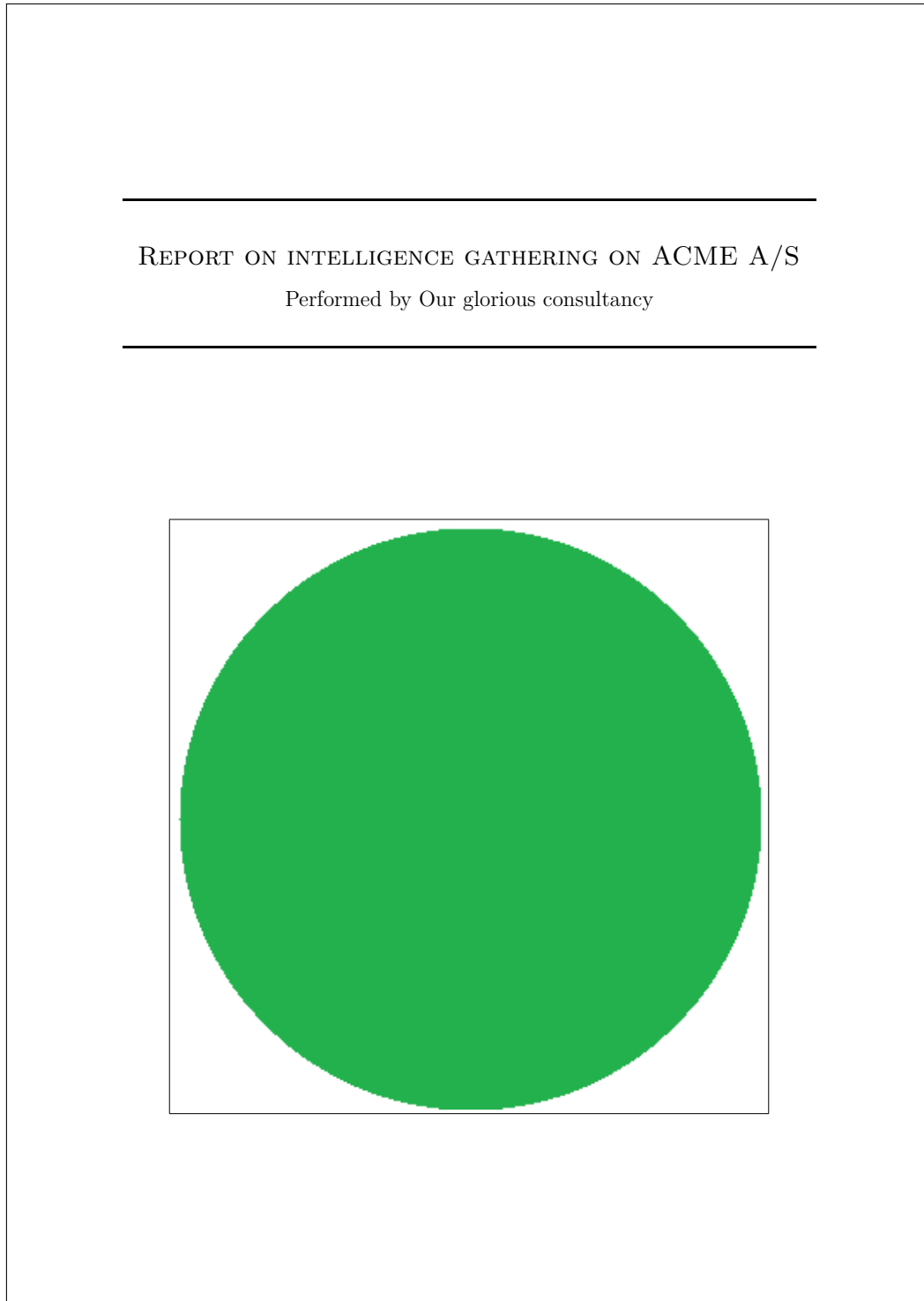
Control/policy/rule within this standard/guideline	Violated?
No organizational details on 3rd party sites (of policies, infrastructure, contact information)	False
No info on organizational structure or job positions	True
No name/phone/email on employees	True
Only use generic emailaddresses publicly (no personal accounts)	False
No critical personal identifiers (Employee no., social security no., D.O.B., “mothers maiden name” and similar)	True

Table 13: The controls/policies/rules of the standard/guideline “Mitnick’s guidelines” and the findings of the intelligence gathering considered in violation of them.

- ‘Lars Estes Henriksen’ (*found from/in/on “ESTES”*)
- The control/policy/rule ‘**No name/phone/email on employees**’ is considered violated based on the following findings:
 - ‘Lars Estes Henriksen’ (*found from/in/on “ESTES”*)
- ‘**Only use generic emailaddresses publicly (no personal accounts)**’ is not considered violated based on the findings.
- The control/policy/rule ‘**No critical personal identifiers (Employee no., social security no., D.O.B., “mothers maiden name” and similar)**’ is considered violated based on the following findings:
 - ‘Nissan X-Trail DIG-T 163 SUV 2WD 6 M/T’ (*found from/in/on “ESTES”*)

B.4 Example of a minimal, auto-generated report

Here follows an example of an auto-generated report generated using the first 10 entries from the file depicted in Appendix D.2. Only one, single finding were assigned only one label to enable a comparison with the “full” example of Appendix B.3.



Intelligence gathering on ACME A/S

“Summer” 2017

1 Exective summary

Based on this report detailing the findings of an *Open Source Intelligence* gathering performed on ACME A/S, it is found that **ACME A/S is vulnerable to 1 of 5 common, OSINT-enabled cyber attack scenarios** reviewed and **violates 1 of 8 standards and guidelines**, which are expected to be applicable to ACME A/S as an organization operating in Denmark. This results in ● severity, which is a good result!

10 findings from the OSINT-gathering were considered for this report.

The conclusions in this report is drawn from a number of commonly occurring scenarios and standards used and may not apply to ACME A/S directly. The results should be considered in a larger context with respect to the overall security maturity of ACME A/S and the risk appetite. Instead the results can be used to – in a simple way – understand the context in which the findings of the OSINT-gathering resides and enhance the understanding and procedures around OSINT-data and its influence on ACME A/S in daily business operations.

Contents

1	Executive summary	2
2	Introduction	4
3	Data found	4
3.1	Statistics on findings	4
3.1.1	Employee	5
3.1.2	SoMe	5
3.1.3	Non-personal internal	5
3.1.4	Supplier	5
3.1.5	Customer	5
4	Scenarios	6
4.1	Spear-phishing	6
4.1.1	Summary of findings	7
4.1.2	Individual requirements	7
4.2	In-person attacks	8
4.2.1	Summary of findings	8
4.2.2	Individual requirements	9
4.3	CEO-fraud	9
4.3.1	Summary of findings	9
4.3.2	Individual requirements	10
4.4	Subverting the supply chain	10
4.4.1	Summary of findings	10
4.4.2	Individual requirements	11
4.5	Targeted (D)DoS	11
4.5.1	Summary of findings	11
4.5.2	Individual requirements	12
5	Standards	13
5.1	Federal CIO Council	13
5.1.1	Individual controls/policies/rules	14
5.2	DS/ISO 27001 – Direct violations	14
5.2.1	Individual controls/policies/rules	15
5.3	Mitnick’s guidelines	15
5.3.1	Individual controls/policies/rules	15

2 Introduction

This report is auto-generated from the findings (data) of a Maltego-investigation performed by Our glorious consultancy towards the company ACME A/S.

The findings come from a gathering of *open source intelligence* (OSINT). OSINT is *all* publicly available information found across many freely available sources – it may be *footprints* of the organization and its employee's daily operations (e.g. from public registers (government or 3rd party)), a product of use of IT systems, web content (e.g. articles, documents and their meta-data), news or active information sharing by individual employees on e.g. social media and fora. Some of the data are avoidable, some are not, but their value to an attack cannot be known until it enters a greater context of an attacker's knowledge and intentions.

To find the information, the attacker can use search engines like Google and Shodan, but also the organizations' own sites, government sites or public registries. The information found is then utilized to try to exploit human psychological mechanisms (i.e. "social engineering") to e.g. establish context with an employee s.t. they place an unmerited degree of trust on an object/subject (e.g. a received e-mail or a person addressing them).

The report suggests how the data found relates to a range of common, targeted cyber attack scenarios enabled by OSINT-data as well as applicable guidelines to organizations acting under Danish legislation.

The scenarios and guidelines are chosen based on the analysis made in master thesis on the subject on DTU Compute summer 2017.

The report is organized into three parts:

- Section 3 categorizes the input-data into 5 different primary categories of information. In each subsection, the sublabels per primary category are listed as well as the count of the findings categorized under each sublabel.
- Section 4 lists 5 common OSINT-enabled cyber attack scenarios which the input-data are considered against. Each scenario is put into a real-world context with an explanation of the scenario and which OSINT-data can go into enabling an attacker to exploit it. For each scenario, it can be seen if ACME A/S are presumed to be vulnerable to the scenarios based on the findings. Additionally we list the input-data, which were found to be contributing to the specific requirements deemed to enable such an attack.
- Section 5 lists 8 standards and guidelines, which are expected to be applicable to the operations of ACME A/S as a Danish organization. For each standard/guideline the policies/controls pertaining to findings such as those appearing here, are listed. If these are violated based on the findings, this is shown with the findings violating.

3 Data found

This section lists the data input to this report. The data is grouped into five categories, each having a number of subcategories which is used to recognize the data in relation to the common cyber attack scenarios and the guidelines covered by this report.

3.1 Statistics on findings

In the tables following, statistics of the findings made during the investigation is given. A bar graph giving an overview of the data within the 5 categories of information, is found in Figure 1.

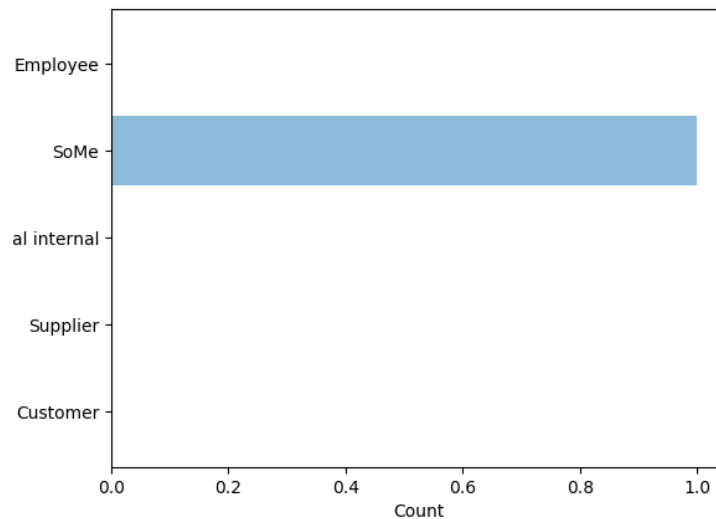


Figure 1: The distribution of findings within the 5 categories of information used in this report.

3.1.1 Employee

There are no findings within this category.

3.1.2 SoMe

Data label	Count
Social Media-accounts (all sites)	1

Table 1: The count of all findings within the category SoMe

3.1.3 Non-personal internal

There are no findings within this category.

3.1.4 Supplier

There are no findings within this category.

3.1.5 Customer

There are no findings within this category.

4 Scenarios

This section relates the findings of Section 3 to 5 common OSINT-enabled cyber attack scenarios.

The scenarios are created as a part of a master thesis project. They seek to cover a wide variety of OSINT-enabled attacks, but is important to understand that it is impossible to describe *all* attack scenarios enabled by the findings used as input to this auto-generated report. These attacks (and the persons behind) employ a vast range of knowledge and information; if some information were not acquired during the research, a specific variety of a scenario could go completely overlooked.

The findings in this section should instead be used as guideline to understand which circumstances or specific data can enable a OSINT-enabled attack against ACME A/S.

5 common cyber attack scenarios were considered. ACME A/S are considered vulnerable to them as shown in Table 2. Details of each of the 5 scenarios and the findings that lead to this conclusion, are found in the following subsections considering each scenario individually.

Cyber attack scenario	Vulnerable?
Spear-phishing	False
In-person attacks	True
CEO-fraud	False
Subverting the supply chain	False
Targeted (D)DoS	False

Table 2: The cyber attack scenarios considered in this report and whether ACME A/S are considered vulnerable towards each of them.

4.1 Spear-phishing

Carried out mostly through emails as the easiest attack vector, but also phone calls, face-to-face or through other means of communication (as people may recognize voice or biometrics of the impersonated person/organization); also called *pretexting*.

The goal is information disclosure for further attacks, directly for e.g. monetary gain (through encouraging bank transfers, acquiring passwords, (bank) account information or NemID-keys) or delivery of a attack payload for e.g. espionage or activism or any other goal.

The most important differences from un-targeted phishing attacks, is that they target a few, specific receivers, put more work into creating a credible email/relation through language, logo's, current activities/contacts of the organization and non-threatening content. However, while they may seek to imitate language of e.g. a professional email or invoice, another trait used in the emails is a sense of urgency and/or secrecy to convince the receiver to perform the task fast (e.g. a bank transfer) and without disclosing anything to colleagues.

To improve credibility, the attacker can employ OSINT to discover:

- Current professional relations (e.g. suppliers, collaborators or customers) found on e.g. LinkedIn, Facebook, public forums, job advertisements (describing technical qualifications needed of new hires) or homepage of the organization or their vendors/customers.
- Private relations or economic interests found on the aforementioned sources or through e.g. public leak data including company domain email addresses.
- Employee names and private e-mail addresses (from e.g. social media accounts) to deliver a malicious payload circumventing organizational countermeasures.
- Specifics of the organization's structure from e.g. informative organizational chart, job postings, points of contact (for homepage, support or legal) or meta-data from documents

Intelligence gathering on ACME A/S

"Summer" 2017

on the organization's homepage. Specifics can include names, positions, job titles, phone numbers, location (e.g. for using the target's national language or TLD) etc. Phone numbers can also serve as an alternative contact medium, where the attacker will then employ other parts of the collected OSINT.

- Knowledge of organizational operations (in addition to the previously mentioned) like travel plans, current issues (from public forums or bug reports).

The attacker can of course also employ technical solutions to increase credibility by e.g. acquiring access to email servers. This requires prior use of social engineering to gather passwords, deliver malicious payloads or similar.

Examples of attack are invoice fraud with fake invoices looking to come from real vendors, coaxing employees into sending money to "colleagues" or trying to get further information on the organization/employees. The most repeated advice in government guidance to hinder these kinds of attacks, are to implement specific procedures of double-checking e.g. money transfers and information disclosures by calling the responsible or the sender and general vigilance of employees.

4.1.1 Summary of findings

Overall it is found that ACME A/S is not vulnerable to this common social engineering attack scenario.

This scenario consists of 6 requirements of which it is expected that 3 of the requirements need to be satisfied in order for ACME A/S to be vulnerable.

During the investigation on ACME A/S, data satisfying 1 of the scenarios' requirements were identified.

In Table 3 each requirement for the scenario "Spear-phishing" is listed. In the following subsections, all data found to be contributing to satisfying the specific attack scenario requirement are listed, s.t. it is possible to gain an insight into exactly what piece of OSINT-data contributed to these findings.

Requirements for this scenario	Satisfied?
Employee names/position	False
Organizational structure	False
Supplier/customer or other professional relations	False
Personal/corporate email addresses, phone numbers	True
Relations (friends, hobbies)	False
Typosquatting domains	False

Table 3: The individual requirements for the cyber attack scenario "Spear-phishing" and the findings of the intelligence gathering satisfying them.

The individual requirements satisfied are considered so by the following findings:

4.1.2 Individual requirements

- 'Employee names/position' is not considered satisfied based on the findings.
- 'Organizational structure' is not considered satisfied based on the findings.
- 'Supplier/customer or other professional relations' is not considered satisfied based on the findings.

Intelligence gathering on ACME A/S

“Summer” 2017

- ‘Personal/corporate email addresses, phone numbers’ is considered satisfied due the following findings:
 - ‘Renault Captur dCi 90’ (*found from/in/on “AB12345”*)
- ‘Relations (friends, hobbies)’ is not considered satisfied based on the findings.
- ‘Typosquatting domains’ is not considered satisfied based on the findings.

4.2 In-person attacks

If the attacker is willing to interact directly with the target/human sources of information in general by e.g. appearing physically on location or calling, an even wider range of scenarios are possible. These are naturally targeted in nature, as the attacker must choose some specific organization/place to appear physically.

Most scenarios described built upon spear-phishing attacks, but requires human interaction, methodically planning and agility of the attack plan. The attacks are diverse in their necessity of information required to work, but all exploit the human mind (i.e. *social engineering*) by different methods.

An example describes how the attacker through human interaction by phone only acquires internal hostnames, credentials to these, out-of-office voice-mails, phone (with internal extensions) and fax numbers, VPN-access and in the end, the data of some project. For this attack, the information found from OSINT-sources beforehand was only:

- Some personal data to verify with (date of birth, family info, social security number etc.).
- Employees of different departments (only a few were necessary, the rest can be offered by the employees called by *namedropping*).
- Company locations/sites.

The rest of the information were discovered during the course of the attack. It should however be noted, that this story involves violation of many policies implemented in modern organization with controls such as those in DS/ISO 27001; those might however fail if the awareness among the employees are not sufficient.

Another method could be for the attacker to show up on premises, which requires proper attire of employees, vendors, shipping handlers etc., and maybe some knowledge of company behavior or locations; afterwards he can use social engineering-techniques to recover the necessary information.

Attacks of this type not requiring any particular OSINT-data includes baiting with infectious USB-devices dropped on the organization’s parking/grounds, tailgating (following employees) inside the organization’s premises or *dumpster diving* to recover confidential information.

4.2.1 Summary of findings

Overall it is found that ACME A/S is vulnerable to this common social engineering attack scenario.

This scenario consists of 5 requirements of which it is expected that 2 of the requirements need to be satisfied in order for ACME A/S to be vulnerable.

During the investigation on ACME A/S, data satisfying 2 of the scenarios’ requirements were identified.

In Table 4 each requirement for the scenario “In-person attacks” is listed. In the following subsections, all data found to be contributing to satisfying the specific attack scenario requirement are listed, s.t. it is possible to gain an insight into exactly what piece of OSINT-data contributed to these findings.

The individual requirements satisfied are considered so by the following findings:

Intelligence gathering on ACME A/S

"Summer" 2017

Requirements for this scenario	Satisfied?
Other personal identifying information	True
Other employee names (name dropping)	True
Internal contact information	False
Internal infrastructure ((host-)names, type, IP's etc.)	False
Supplier information (names, relation to organization)	False

Table 4: The individual requirements for the cyber attack scenario "In-person attacks" and the findings of the intelligence gathering satisfying them.

4.2.2 Individual requirements

- 'Other personal identifying information' is considered satisfied due the following findings:
 - 'Renault Captur dCi 90' (*found from/in/on "AB12345"*)
- 'Other employee names (name dropping)' is considered satisfied due the following findings:
 - 'Renault Captur dCi 90' (*found from/in/on "AB12345"*)
- 'Internal contact information' is not considered satisfied based on the findings.
- 'Internal infrastructure ((host-)names, type, IP's etc.)' is not considered satisfied based on the findings.
- 'Supplier information (names, relation to organization)' is not considered satisfied based on the findings.

4.3 CEO-fraud

Considered a specific kind of spear-phishing, this attack impersonates or targets C-level employees – also called *whales*, as they are "the big targets".

The aim is to perform acts similar to spear-phishing, but due to the large amount of money that may be involved with C-level roles, a larger reward can be collected by the attacker.

Prerequisites and traits of the attack are similar to spear-phishing as well; it may however *not be necessary to know any vendors/customers of the organization* for this attack, but only:

- Name of head of the company.
- His e-mail address (to mimic or create something similar).
- Managers/employees authorized to perform a transfer of funds.

The emails may be even better crafted than regular spear-phishing emails through e.g. more formal/correct language.

Examples of *spear-phishing* can also be considered whaling; in a specific example, The National Museum of Art in Denmark were recently phished for 805.000 DKK by impersonating the CEO and targeting an employee with privileged access to the accounts.

4.3.1 Summary of findings

Overall it is found that ACME A/S is not vulnerable to this common social engineering attack scenario.

Intelligence gathering on ACME A/S

“Summer” 2017

This scenario consists of 3 requirements of which it is expected that 2 of the requirements need to be satisfied in order for ACME A/S to be vulnerable.

During the investigation on ACME A/S, data satisfying 0 of the scenarios’ requirements were identified.

In Table 5 each requirement for the scenario “CEO-fraud” is listed. In the following subsections, all data found to be contributing to satisfying the specific attack scenario requirement are listed, s.t. it is possible to gain an insight into exactly what piece of OSINT-data contributed to these findings.

Requirements for this scenario	Satisfied?
Name/e-mail of CEO/CFO	False
Name/e-mail on (privileged) employee	False
Name/e-mail of some partner	False

Table 5: The individual requirements for the cyber attack scenario “CEO-fraud” and the findings of the intelligence gathering satisfying them.

The individual requirements satisfied are considered so by the following findings:

4.3.2 Individual requirements

- ‘Name/e-mail of CEO/CFO’ is not considered satisfied based on the findings.
- ‘Name/e-mail on (privileged) employee’ is not considered satisfied based on the findings.
- ‘Name/e-mail of some partner’ is not considered satisfied based on the findings.

4.4 Subverting the supply chain

This is to attack equipment or software being delivered to the organization. The goal is to deliver a payload through the regular supply chain of the organization; suppliers which the organization has already put a high level of trust in and perhaps thus are less likely to question deliveries/content from.

We know NSA performs this practice against hardware/servers exported from the US and some believe that Huawei-equipment does the same. In a specific attack, a vendor of scanners running Microsoft XP Embedded OS were shipped with malware. The malware targeted ERP-systems of shipping and logistics and later also manufacturers.

A lot of information may have gone into compromising the manufacturer himself, but from the actual target organization, it may only necessary to know:

- A type of software used in the company (e.g. an ERP-system).
- A type of hardware deployed or the distributor bought from.
- The attack could also be leveraged by identifying the employee responsible of procurement of IT equipment (and contact information) e.g. from an organizational chart of social media.

4.4.1 Summary of findings

Overall it is found that ACME A/S is not vulnerable to this common social engineering attack scenario.

Intelligence gathering on ACME A/S

“Summer” 2017

This scenario consists of 3 requirements of which it is expected that 2 of the requirements need to be satisfied in order for ACME A/S to be vulnerable.

During the investigation on ACME A/S, data satisfying 0 of the scenarios’ requirements were identified.

In Table 6 each requirement for the scenario “Subverting the supply chain” is listed. In the following subsections, all data found to be contributing to satisfying the specific attack scenario requirement are listed, s.t. it is possible to gain an insight into exactly what piece of OSINT-data contributed to these findings.

Requirements for this scenario	Satisfied?
Employee responsible of procurement (or other specific department)	False
Some software used in the organization (and the supplier)	False
Some hardware used in the organization (and the supplier)	False

Table 6: The individual requirements for the cyber attack scenario “Subverting the supply chain” and the findings of the intelligence gathering satisfying them.

The individual requirements satisfied are considered so by the following findings:

4.4.2 Individual requirements

- ‘**Employee responsible of procurement (or other specific department)**’ is not considered satisfied based on the findings.
- ‘**Some software used in the organization (and the supplier)**’ is not considered satisfied based on the findings.
- ‘**Some hardware used in the organization (and the supplier)**’ is not considered satisfied based on the findings.

4.5 Targeted (D)DoS

This attack can be employed if the attacker is politically motivated and wants to shut down a service/website/operations of the organization, but also as a tool of extortion. The danger of this is inherent in servers connected to the Internet; hostnames are quick to resolve and target, but if public IP’s not meant to be exposed/used by regular users, are found through e.g. Shodan or public pastes of stolen data, the right security measures might not be present.

It can also be a problem if internal IP-addresses are found from e.g. internal documents or in website descriptions, as it can be used to claim credibility (by proving knowledge of internal network components). Additionally, techniques exist to route traffic through public IP’s to unintended servers inside the network.

4.5.1 Summary of findings

Overall it is found that ACME A/S is not vulnerable to this common social engineering attack scenario.

This scenario consists of 2 requirements of which it is expected that 1 of the requirements need to be satisfied in order for ACME A/S to be vulnerable.

During the investigation on ACME A/S, data satisfying 0 of the scenarios’ requirements were identified.

In Table 7 each requirement for the scenario “Targeted (D)DoS” is listed. In the following subsections, all data found to be contributing to satisfying the specific attack scenario requirement

Intelligence gathering on ACME A/S

“Summer” 2017

are listed, s.t. it is possible to gain an insight into exactly what piece of OSINT-data contributed to these findings.

Requirements for this scenario	Satisfied?
Internal IP's or hostnames	False
External IP's	False

Table 7: The individual requirements for the cyber attack scenario “Targeted (D)DoS” and the findings of the intelligence gathering satisfying them.

The individual requirements satisfied are considered so by the following findings:

4.5.2 Individual requirements

- ‘**Internal IP's or hostnames**’ is not considered satisfied based on the findings.
- ‘**External IP's**’ is not considered satisfied based on the findings.

Intelligence gathering on ACME A/S

“Summer” 2017

5 Standards

This section relates the findings of Section 3 to 8 standards and guidance applicable to ACME A/S as an organization operating under Danish legislation.

The standards and guidance are identified through a master thesis project, where industry standards and -guidelines from reknowned institutions were considered. In particular, material applicable to Danish organizations were considered.

The material chosen are published by government bodies as NCSC (UK), Federal CIO Council (US) and CFCS (DK) and standardization groups as NIST (US) and the ISO-group. Finally, industry veteran Kevin Mitnick’s guidance are used.

The content of this section aim to *give guidance* to ACME A/S to understand the findings in context of the standards and guidance that are likely to influence the daily operations of the organization as imposed by legislation or in other ways.

The stanarads and guidance in turn can help shed a different light on how specific data (under some circumstances) or can enable a OSINT-enabled attack against ACME A/S.

8 standards/guidance were considered – each consisting of a number of controls/policies/rules. ACME A/S are considered vulnerable to a standard/guideline of one controls/policies/rules are violated. We consider ACME A/S in violation as shown in Table 8.

Details of each of the 8 standards/guidance and the findings that lead to this conclusion, are found in the following subsections considereing each standard individually¹

Standard/guideline	Violated?
NIST	False
NCSC	False
CFCS	False
CPNI	False
Agency for Digitisation	False
Federal CIO Council	True
DS/ISO 27001 – Direct violations	False
Mitnick’s guidelines	False

Table 8: The standards/guidance considered in this report and whether ACME A/S are considered in violation of them.

5.1 Federal CIO Council

Overall it is found that ACME A/S is in violation of this standard/guideline.

This standard/guideline consists of 2 individual controls/policies/rules; if one of these are violated, we consider ACME A/S in violation of this particular standard/guideline.

During the investigation on ACME A/S, data satisfying 1 of the controls/policies/rules were identified.

In Table 9 each control/policy/rule for the standard/guideline “Federal CIO Council” is listed. In the following subsections, all data found to be contributing to violating the specific control/policy/rule are listed, s.t. it is possible to gain an insight into exactly what piece of OSINT-data contributed to these findings.

¹Please note that standard/guideline 1-5 are not reflected in the current implementation of the code generating

Intelligence gathering on ACME A/S

“Summer” 2017

Control/policy/rule within this standard/guideline	Violated?
Separate professional and personal life: Don't use corp. email addresses for personal accounts (SoMe, school contact sheet etc.) – especially not in relation with other personal details	True
Present yourself properly online (including not disclosing valuable information to an adversary)	False

Table 9: The controls/policies/rules of the standard/guideline “Federal CIO Council” and the findings of the intelligence gathering considered in violation of them.

The individual controls/policies/rules are considered violated by the following findings:

5.1.1 Individual controls/policies/rules

- The control/policy/rule ‘**Separate professional and personal life: Don't use corp. email addresses for personal accounts (SoMe, school contact sheet etc.) – especially not in relation with other personal details**’ is considered violated based on the following findings:
 - ‘Renault Captur dCi 90’ (*found from/in/on “AB12345”*)
- ‘**Present yourself properly online (including not disclosing valuable information to an adversary)**’ is not considered violated based on the findings.

5.2 DS/ISO 27001 – Direct violations

Overall it is found that ACME A/S is not in violation of this standard/guideline.

This standard/guideline consists of 5 individual controls/policies/rules; if one of these are violated, we consider ACME A/S in violation of this particular standard/guideline. During the investigation on ACME A/S, data satisfying 0 of the controls/policies/rules were identified.

In Table 10 each control/policy/rule for the standard/guideline “DS/ISO 27001 – Direct violations” is listed. In the following subsections, all data found to be contributing to violating the specific control/policy/rule are listed, s.t. it is possible to gain an insight into exactly what piece of OSINT-data contributed to these findings.

Control/policy/rule within this standard/guideline	Violated?
Protect personal identifying information	False
Records shall be protected from unauthorized access	False
Transfer policies/controls for data shall be in place	False
Procedures for handling classified information shall be implemented (i.e. classified information should not be publicly available)	False
Information involved in electronic messaging shall be appropriately protected	False

Table 10: The controls/policies/rules of the standard/guideline “DS/ISO 27001 – Direct violations” and the findings of the intelligence gathering considered in violation of them.

The individual controls/policies/rules are considered violated by the following findings:

this report.

Intelligence gathering on ACME A/S

"Summer" 2017

5.2.1 Individual controls/policies/rules

- **'Protect personal identifying information'** is not considered violated based on the findings.
- **'Records shall be protected from unauthorized access'** is not considered violated based on the findings.
- **'Transfer policies/controls for data shall be in place'** is not considered violated based on the findings.
- **'Procedures for handling classified information shall be implemented (i.e. classified information should not be publicly available)'** is not considered violated based on the findings.
- **'Information involved in electronic messaging shall be appropriately protected'** is not considered violated based on the findings.

5.3 Mitnick's guidelines

Overall it is found that ACME A/S is not in violation of this standard/guideline.

This standard/guideline consists of 5 individual controls/policies/rules; if one of these are violated, we consider ACME A/S in violation of this particular standard/guideline.

During the investigation on ACME A/S, data satisfying 0 of the controls/policies/rules were identified.

In Table 11 each control/policy/rule for the standard/guideline "Mitnick's guidelines" is listed. In the following subsections, all data found to be contributing to violating the specific control/policy/rule are listed, s.t. it is possible to gain an insight into exactly what piece of OSINT-data contributed to these findings.

Control/policy/rule within this standard/guideline	Violated?
No organizational details on 3rd party sites (of policies, infrastructure, contact information)	False
No info on organizational structure or job positions	False
No name/phone/email on employees	False
Only use generic emailaddresses publicly (no personal accounts)	False
No critical personal identifiers (Employee no., social security no., D.O.B., "mothers maiden name" and similar)	False

Table 11: The controls/policies/rules of the standard/guideline "Mitnick's guidelines" and the findings of the intelligence gathering considered in violation of them.

The individual controls/policies/rules are considered violated by the following findings:

5.3.1 Individual controls/policies/rules

- **'No organizational details on 3rd party sites (of policies, infrastructure, contact information)'** is not considered violated based on the findings.
- **'No info on organizational structure or job positions'** is not considered violated based on the findings.
- **'No name/phone/email on employees'** is not considered violated based on the findings.

Intelligence gathering on ACME A/S

“Summer” 2017

- ‘Only use generic emailaddresses publicly (no personal accounts)’ is not considered violated based on the findings.
- ‘No critical personal identifiers (Employee no., social security no., D.O.B., “mothers maiden name” and similar)’ is not considered violated based on the findings.

Appendix C

CPNI hostile reconnaissance checklist

This is the checklist of questions security managers can use to get a picture of their vigilance towards hostile reconnaissance. The right security can counter many attack scenarios, as they all rely on gaining some vital information about their target beforehand; despite different scenarios, much of the necessary information can be identical.

The checklist is to be used when the security managers evaluate the organization's security after reading [11] and understanding the principles of *deny*, *detect* and *deter*. They should consider the six themes and ask the questions as part of this.

The six themes of the checklist are:

- Secure online presence
- Robust entry process
- Hostile reconnaissance threat is understood
- Strong staff security awareness
- Vigilant and professional security
- Deterrence strategy

Question	What will be the result?
Secure online presence	
Does your organization think about the information it puts into the public domain and consider what positive/negative impact this may have on those engaged in hostile reconnaissance?	Your organization considers and manages what information is available about it in the public domain and this will help deter those carrying out online hostile reconnaissance

CPNI hostile reconnaissance checklist

Do your employees understand why they need to be aware of what information they reveal about themselves or their organisation when online?	Your employees consider the impact their digital footprint has on both them and the organisation they work for, thereby making it more difficult for hostiles to harvest information from them.
Does your organisation understand the threat posed by employees inadvertently giving away information or allowing unauthorised access or malicious software onto your systems?	Your organisation has an understanding of how spear phishing (and similar) attacks are conducted and what can be done to mitigate them.
Robust entry process	
Do your employees undergo identity and document verification training?	Employees tasked with document verification, whether during pre-employment screening and/or during visitor entry, are vigilant to the threat of fraudulent documentation.
Are your security personnel sufficiently motivated to identify, deter or detect hostile reconnaissance?	Motivated, attentive and observant security personnel that can form a highly-effective deterrent presence and final line of defence where other interventions may have failed.
Hostile reconnaissance threat is understood	
Do you understand what hostile reconnaissance is, where it may be conducted at your site and what you can do to deter or detect it?	Potential hostile reconnaissance points are identified and mitigation measures introduced.
Do you make use of deterrence materials such as security posters aimed at hostiles, in and around your site?	Security managers are given the materials and support to carry out a deterrence messaging campaign, resulting in the deterring or detecting of hostiles.
Strong staff security awareness	
Have you measured your organisation's security culture?	Your organisation understands its security culture and identifies where and why it might need to change.
Do your employees know why they need to be vigilant in and around their place of work?	Employees display vigilant behaviours in and around the workplace, thereby making them less of a target and more likely to identify those conducting hostile reconnaissance.
Have your employees been educated as to why their security behaviours in the workplace matter?	Employees display good security behaviours in and around the workplace.

CPNI hostile reconnaissance checklist

Do your employees know what social engineering looks like and what to do if they think it is happening to them?	Employees recognise social engineering approaches and respond appropriately.
Do your employees understand why they need to be aware of what information they reveal about themselves or their organisations online?	Your employees consider the impact their digital footprint has on both them and the organisation they work for, thereby making it more difficult for hostiles to harvest information from them.
Do your organisation's line managers understand the role they have to play in security?	Managers consider security while making day-to-day business decisions and ensure their teams are kept up-to-date on security matters.
Vigilant and professional security	
Does your security department understand the threats it faces?	Security personnel understand the threats posed to their organisation and are motivated to identify and disrupt hostile reconnaissance.
Do your security personnel display a professional-looking presence, profile and posture?	Security officers are motivated to identify and disrupt hostile reconnaissance.
Have your security personnel received training in detecting suspicious behaviour and tactical questioning?	Security officers can more readily identify hostile reconnaissance and resolve suspicions through questioning.
Do your CCTV operators know what to look for in terms of hostile reconnaissance?	Improved effectiveness of CCTV operators in deterring and detecting hostile reconnaissance.
Deterrence strategy	
Do you make use of deterrence materials in and around your site?	Hostiles are deterred by, or detected as a result of, your deterrence materials.
Are you considering how all the elements of your security and communications assets can be used together when deterring and detecting hostile reconnaissance? Are you intelligently promoting your security measures?	Security managers understand the threat from hostile reconnaissance. Your organisation's security assets are coordinated and utilised to create the maximum effect.

Table C.1: The checklist from CPNI's guidance on countering hostile reconnaissance. From [11].

Appendix D

Figures

Figures

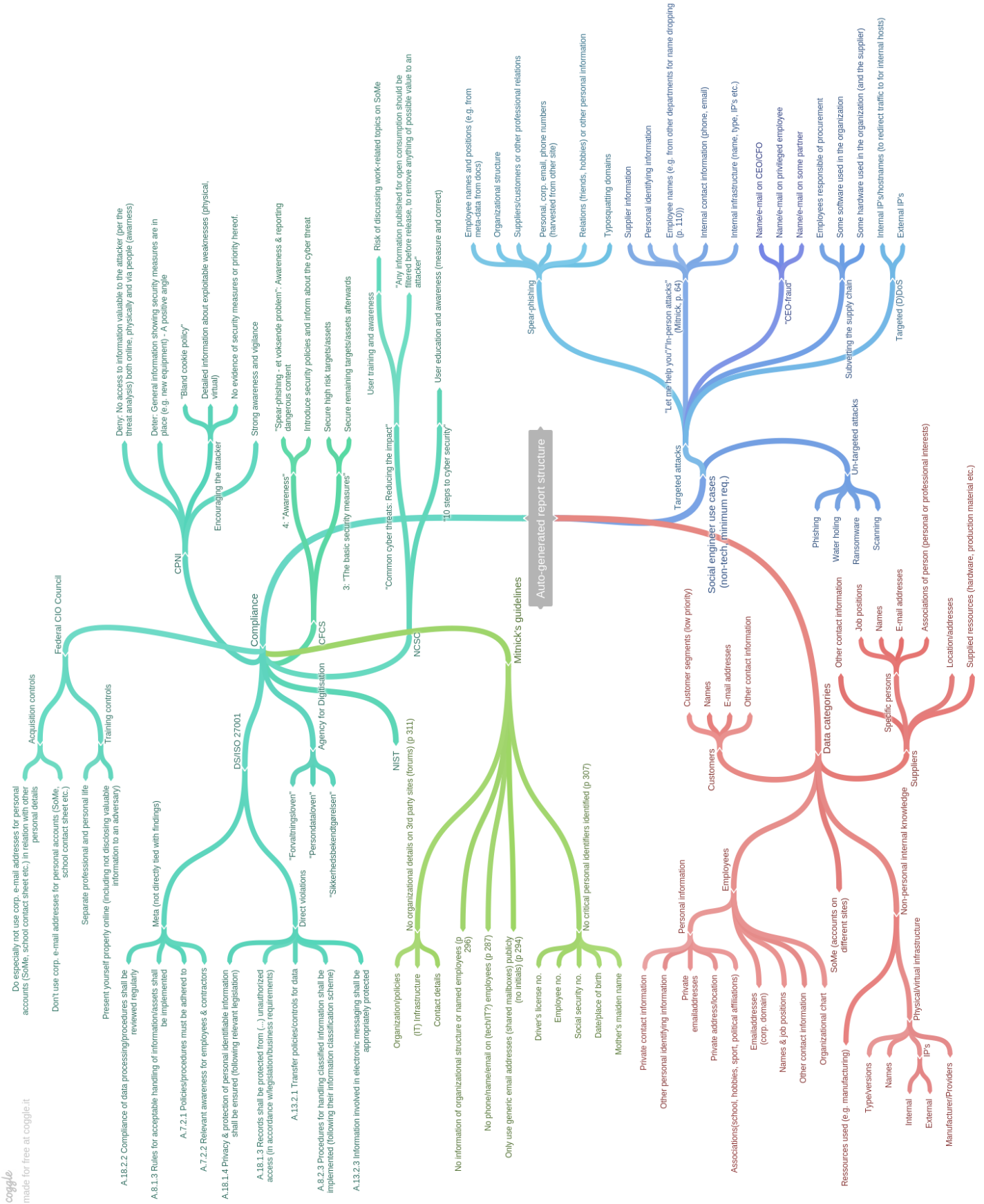


Figure D.1: The mindmap for the content of the auto-generated report.

D.1 nrpla.de field implementation

Read value: BB29177 and type thesis.licenseplate (from entity "BB29177")	
#### Data from basic search #### (from entity "BB29177")	
x key: axles	value: 2 (from entity "BB29177")
x key: last_inspection_odometer	value: 130000 (from entity "BB29177")
x key: engine_power	value: None (from entity "BB29177")
x key: color	value: {u'id': 1, u'name': u'Ukendt'} (from entity "BB29177")
x key: vin	value: TMBHC26Y154335271 (from entity "BB29177")
x key: last_inspection_result	value: Godkendt (from entity "BB29177")
key: body_type	value: None (from entity "BB29177")
key: type_approval_code	value: (from entity "BB29177")
x key: total_weight	value: 1600 (from entity "BB29177")
x key: registration	value: BB29177 (from entity "BB29177")
key: vehicle_id	value: 1024201200516223 (from entity "BB29177")
x key: top_speed	value: None (from entity "BB29177")
x key: vehicle_weight	value: 1100 (from entity "BB29177")
x key: fuel_type	value: Benzin (from entity "BB29177")
x key: fuel_efficiency	value: 15.4 (from entity "BB29177")
x key: towing_weight	value: 450 (from entity "BB29177")
x key: version	value: 1,4 COMBI (from entity "BB29177")
key: expire_date	value: None (from entity "BB29177")
key: wheels	value: (from entity "BB29177")
x key: first_registration_date	value: 2005-09-02 (from entity "BB29177")
x key: type	value: Personbil (from entity "BB29177")
key: minimum_seats	value: None (from entity "BB29177")
key: engine_displacement	value: None (from entity "BB29177")
key: last_inspection_date	value: 2016-04-11 (from entity "BB29177")
x key: brand	value: Skoda (from entity "BB29177")
key: drive_axles	value: (from entity "BB29177")
x key: use	value: {u'id': 1, u'name': u'Privat personk\&rsel'} (from entity "BB29177")
x key: engine_cylinders	value: None (from entity "BB29177")
x key: model	value: Fabia (from entity "BB29177")
x key: registration_status	value: Registreret (from entity "BB29177")
x key: leasing_period_end	value: None (from entity "BB29177")
x key: model_year	value: None (from entity "BB29177")
key: extra_equipment	value: (from entity "BB29177")
key: technical_total_weight	value: 1600 (from entity "BB29177")
x key: towing_weight_brakes	value: 800 (from entity "BB29177")
x key: status_updated_date	value: 2016-04-12 (from entity "BB29177")
x key: owner_type	value: Privatperson (from entity "BB29177")
x key: leasing_period_start	value: None (from entity "BB29177")
#### Data from DMR search #### (from entity "BB29177")	
x key: insurance_policy_number	value: N/A (from entity "BB29177")
x key: insurance_created	value: 12-04-2016 (from entity "BB29177")

	u'Halv\xe5rlig', u'type': u'Gr\xfa8n Ejerafgift'), {u'amount': u'1.190,00 kr.', u'frequency': u'-' , u'type': u'Sum'}], u'history': [{u'from': u'01-04-2017', u'required': u'Ja', u'to': u'30-09-2017', u'amount': u'1.190,00 kr.', u'determined': u'01-04-2017', u'registration': u'BB29177', u'type': u'Gr\xfa8n Ejerafgift'), {u'from': u'01-10-2016', u'required': u'Ja', u'to': u'31-03-2017', u'amount': u'1.190,00 kr.', u'determined': u'01-10-2016', u'registration': u'BB29177', u'type': u'Gr\xfa8n Ejerafgift'), {u'from': u'12-04-2016', u'required': u'Ja', u'to': u'30-09-2016', u'amount': u'1.117,28 kr.', u'determined': u'12-04-2016', u'registration': u'BB29177', u'type': u'Gr\xfa8n Ejerafgift'), {u'from': u'22-03-2016', u'required': u'Ja', u'to': u'30-04-2016', u'amount': u'257,83 kr.', u'determined': u'22-03-2016', u'registration': u'XS44629', u'type': u'Refution - Gr\xfa8n Ejerafgift'), {u'from': u'01-11-2015', u'required': u'Ja', u'to': u'30-04-2016', u'amount': u'1.190,00 kr.', u'determined': u'01-11-2015', u'registration': u'XS44629', u'type': u'Gr\xfa8n Ejerafgift'), {u'from': u'01-05-2015', u'required': u'Ja', u'to': u'31-10-2015', u'amount': u'1.190,00 kr.', u'determined': u'01-05-2015', u'registration': u'XS44629', u'type': u'Gr\xfa8n Ejerafgift'), {u'from': u'01-11-2014', u'required': u'Ja', u'to': u'30-04-2015', u'amount': u'1.110,00 kr.', u'determined': u'01-11-2014', u'registration': u'XS44629', u'type': u'Gr\xfa8n Ejerafgift'), {u'from': u'01-05-2014', u'required': u'Ja', u'to': u'31-10-2014',
x key: taxes	
x key: insurance_company	value: TRYG FORSIKRING A/S (from entity "BB29177")
key: type_approval_code	value: E13556-05 (from entity "BB29177")
x key: insurance_status	value: Aktiv (from entity "BB29177")
key: automatic	value: 0 (from entity "BB29177")
#### Data from Debt search #### (from entity "BB29177")	
x key: no_debt	value: True (from entity "BB29177")
x key: debtors	value: (from entity "BB29177")
x key: amount	value: 0 (from entity "BB29177")
x key: creditors	value: (from entity "BB29177")
key: date_number	value: (from entity "BB29177")
#### Data from Inspections search ####	
key: category	value: Registreringssyn uden ndring
key: car_type	value: M1-Personbil <= 3500 kg
key: errors	value: []
key: car_brand	value: SKODA
key: car_model	value: FABIA
x key: company	value: Trekantens Bilsyn ApS
key: car_vin	value: TMBHC26Y154335271
x key: cvr	value: 28148895
key: reinspection_date	value:
x key: odometer	value: 130000
x key: location	value: Ladegrdsvej 8B7100 Vejle
x key: time	value: 16:20
x key: date	value: 11-04-2016
x key: car_registration	value: XS44629
key: type	value: Frste syn
key: service_message	value:
x key: result	value: Godkendt

D.2 Sample csv-export from Maltego

AB12345,Renault Captur dCi 90
AB12345,Sabri Elhaj Moussa
ESTES,Catharina Estes Henriksen
ESTES,Lars Estes Henriksen
ESTES,Nissan X-Trail DIG-T 163 SUV 2WD 6 M/T
djoef-dk.mx1.comendosystems.com,89.104.206.4
djoef.dk,djoef-dk.mx1.comendosystems.com
djoef.dk,djoef@djoef.dk
djoef.dk,edit.djoef.dk
djoef.dk,https://www.djoef.dk/~media/documents/djoef/f/forside.ashx?la=da www.djoef.dk
djoef.dk,https://www.djoef.dk/~media/documents/djoef/p/ansttelseskontrakt--privatansat-
engelsk.ashx?la=da Mellem undertegnede - djoef.dk
djoef.dk,Imp@djoef.dk
djoef.dk,ns1.djoef.dk
djoef.dk,www.djoef.dk
djoef@djoef.dk,regionh@regionh.dk
djoef@djoef.dk,via@securemail.via.dk
dr.dk,2018-03-31
dr.dk,D5109-DK
dr.dk,DH4991-DK
dr.dk,DR3450-DK
dr.dk,TD4070-DK
dr.dk,dns101.telia.com
dr.dk,dns102.telia.com
dr.dk,ns01.dr.dk
mtborientering.dk,2018-06-30
mtborientering.dk,Troels Bent Hansen
mtborientering.dk,ns1.unoeuro.com
ninaborrson.dk,2018-03-31
ninaborrson.dk,DA49-DK
ninaborrson.dk,ns.dandomain.dk
ninaborrson.dk,ns2.dandomain.dk
ninaborrson.dk,ns3.dandomain.dk
www.djoef.dk,Djøf: Ledernes Juristernes Økonomernes og Studerendes Fagforbund – Djøf er også en
fagforening og faglig organisation. Tidligere Danmarks Jurist- og Økonomforbund og C3
furesø.dk