Gjøvik University College

Ethical Hacking & Penetration Testing

Information Gathering & Book Review

Assignment #1

Victor Rudolfsson - 120912

October 10, 2014

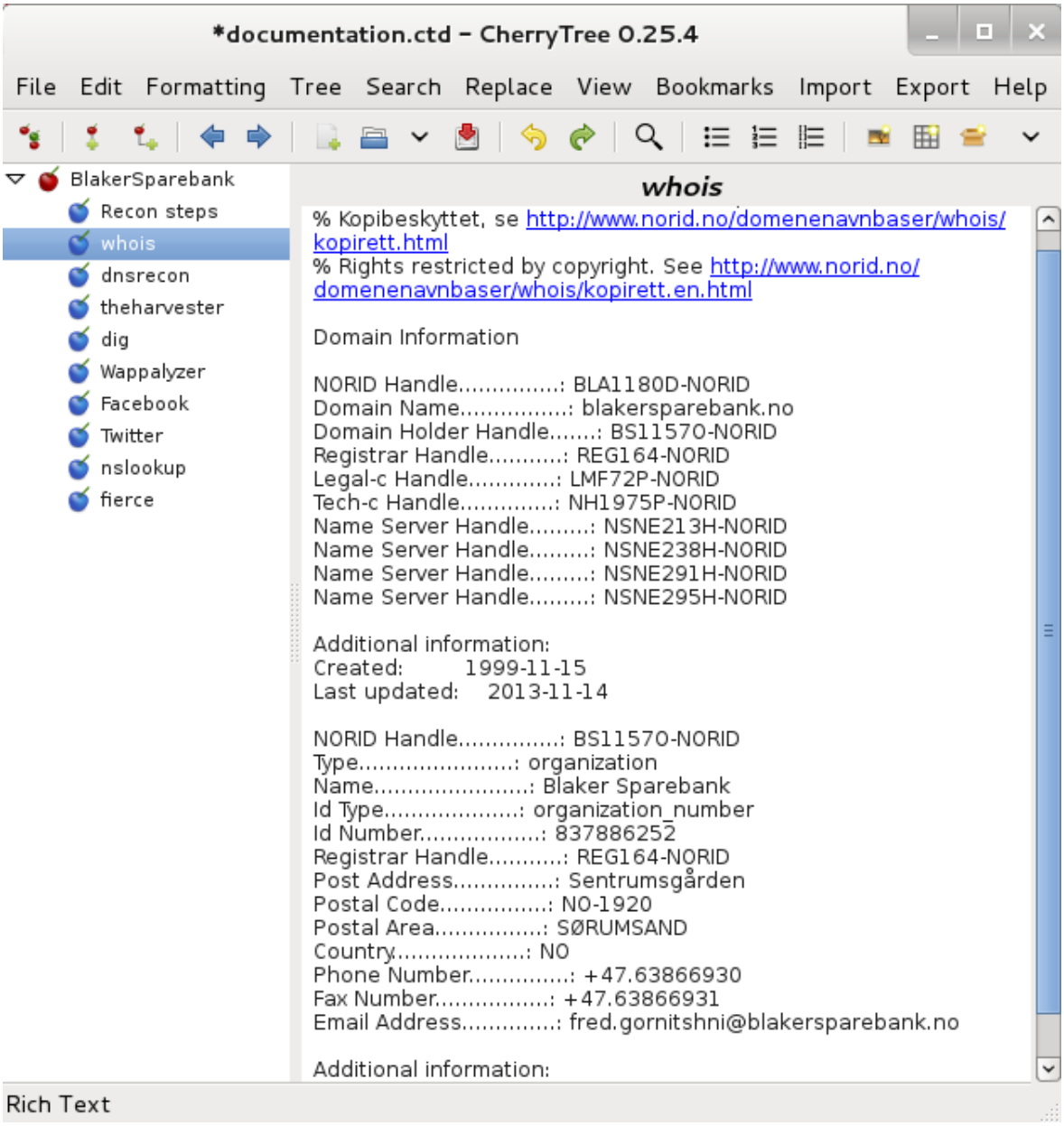
1 Part 1

1.1 Introduction

I was assigned the URL *www.blakersparebank.no* to perform passive reconnaissance on. I began outlining a plan of what I could do without performing any active scans, and came up with the following strategy:

1. Manually crawl website to find information about employees and positions
2. Gather information about each employee into a Maltego graph
3. Check social network presence of employees
4. Identify technologies used by the server hosting the website
5. Look up nameservers & whois-information
6. Check if other sites are hosted on the same server - a house has many doors

For documentation, I used *CherryTree*, a fantastic text editor supporting tree-style notes (see below) and Maltego / Maltego CaseFile. I began with Maltego, but as the graph became more and more about employees, I imported the graph into Maltego CaseFile.



1.2 Reconnaissance

I began scanning for interesting documents (PDFs, XLS, DOC, DOCX, etc) using Google d0rks with *inurl* and/or *format* keys, but with no luck - or rather, with a few

reports on revenue but no information I found interesting. However, the website did have an *About us* section, which provided plenty of information about employees and their position. Using this, I created a plausible hierarchy of the employees (as I interpreted their positions).

I could then proceed to assign node pictures, and fetched the employee images with a short line of bash:

Listing 1: Fetching images

```
~:# for img in $(wget -qO- http://blakersparebank.no/om-oss | \
 grep -oP '\w*\.(JPG|jpg)'); \
 do wget http://blakersparebank.no/~/media/banker/blakersparebank/ansatte/$i
 -o Pictures/$img; done
```

Initially, I was tempted to attempt bruteforcing directories to get a better idea of the filesystem, since the image URLs gave a hint of the directory structure, but refrained from this as I presumed this fell under the category of scanning.

Setting up the Maltego graph took some time, as researching each employee is rather time consuming, but after I was done with this, I proceeded with the next step in the strategy I had outlined – check what technologies the server used. I used *Wappalyzer*, a Firefox (or, in this case, Iceweasel) addon that analyzes what technologies are used on websites by using a bunch of regex scans for signature code in HTML (meta tags/classes/scripts), URLs, response headers and global JavaScript variables, which gives a hint to what technologies may be in use.

Wappalyzer suggested BlakerSparebank was using *Microsoft IIS* as a webserver, the site building on ASP.NET code. This suggests (and so did Wappalyzer) that the server is therefore running Windows Server.

This was the information I could gather from Wappalyzer:

Having fed this information into my graph and CherryTree, I proceeded to look up nameservers using the regular *dig* command.

OS	Windows Server
Web Server	Microsoft IIS
Language	ASP.NET
Scripts:	jQuery
	Modernizr
	New Relic
	yepnope.js

Listing 2: dig'ing through DNS

```
root@battlestation:~# dig blakersparebank.no

; <<>> DiG 9.8.4-rpz2+rl005.12-P1 <<>> blakersparebank.no
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 15445
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;blakersparebank.no.            IN      A

;; ANSWER SECTION:
blakersparebank.no.     60      IN      A       153.110.253.25

;; Query time: 372 msec
;; SERVER: 192.168.0.10#53(192.168.0.10)
;; WHEN: Fri Oct 10 00:27:42 2014
;; MSG SIZE  rcvd: 52
```

This disappointed me greatly, as I had hoped for much more information, but nevertheless using the A-record for the IP I could proceed to perform a reverse DNS. Having first tried this with *fierce* with no luck, I gave *theharvester* a try.

theharvester is a fantastic little program that now comes bundled with Kali as well, and harvests information from all kinds of sources. It searches Google, Bing, PGP,

LinkedIn, People123, Jigsaw, as well as allows for reverse DNS lookup of a domain. It even has support for Shodan. I let it perform a full harvest on all search engines, and a reverse DNS lookup.

Listing 3: theharvester performing full harvest

```
root@battlestation:~# theharvester -d blakersparebank.no -nhvb all

...

Full harvest..
[-] Searching in Google..
        Searching 0 results...
        Searching 100 results...
[-] Searching in PGP Key server..
[-] Searching in Bing..
        Searching 50 results...
        Searching 100 results...
[-] Searching in Exalead..

Searching 150 results...

[+] Emails found:
------------------
post@blakersparebank.no
reidun.skomdal@blakersparebank.no
blaker.sparebank@blakersparebank.no
jho@blakersparebank.no

[+] Hosts found in search engines:
-----------------------------------
153.110.253.25:www.blakersparebank.no
```

```
[+] Starting active queries:
Hosts found after reverse lookup:
--------------------------------
[+] Virtual hosts:
==================
153.110.253.25  eika
153.110.253.25  surnadal-sparebank
153.110.253.25  terra-eiendomsmegling.no
153.110.253.25  tos.no
153.110.253.25  eikabk.no
153.110.253.25  aktiv.no
153.110.253.25  aurskog-sparebank.no
153.110.253.25  twocards.no
153.110.253.25  rorosbanken.no
153.110.253.25  honefossbank.no

...

[+] Shodan Database search:
153.110.253.25:www.blakersparebank.no
        Searching for: 153.110.253.25:www.blakersparebank.no
153.110.253.25:terra-eiendomsmegling.no
153.110.253.25:tos.no
153.110.253.25:eikabk.no
153.110.253.25:aktiv.no
153.110.253.25:aurskog-sparebank.no
153.110.253.25:twocards.no
153.110.253.25:rorosbanken.no
153.110.253.25:honefossbank.no

...

[+] Shodan results:
===================
```

In the output above, I cut off more than half the rows as to not make this report exceed the page limit and be able to continue with it. Although *theharvester* found nothing on Shodan, and only 4 emails, it did find a plethora of domains hosted on the same server!

I began visiting these domains manually to see what they were, as the majority of them seemed to be other banks. They were indeed other banks, with nearly identical websites.

From this, and the previous information collected via Wappalyzer, I could make a qualified guess that these websites are built, run and operated with *Microsoft SharePoint*.

1.3 Summary

The amount of information people share publicly on Facebook never cease to amaze me, and although many employees couldn't be found on Facebook, many could and several of them publicly listed their family members, work, photos, etc. I included most of this information in the Maltego graph, together with addresses, phone numbers, profiles and other information I could locate.

As for the server, I could conclude (fairly accurately I think) that it builds on ASP.NET and Microsoft SharePoint, running on a Microsoft IIS web server on a Windows Server along with plenty of other banks as virtual hosts. The downside of this is that if one of these gets compromised, they all do, as updates are often pushed simultaneously. On the other hand if one of them gets fixed, they all do as well.

1.3.1 Tools used

Most of the tools I used were found in Kali under the *Information Gathering* section's *OSINT* or *DNS* subsections, or tools I knew of previously such as *Wappalyzer* or *V3n0md0rk3r*.

Finally, these were the tools I used:

1. ~~Archive.org~~ - Trying to find useful old (cached) dox (no success)

2. CaseFile - Another version of Maltego, more people oriented
3. CherryTree - A treestyled notepad
4. dig - DNS record lookup
5. ~~dnsrecon~~ - For reverse DNS, but got no useful information from this tool
6. Facebook Graph Search - Finding 3 employee Facebook profiles with *People who work at BlakerSparebank*
7. ~~fierce~~ - Also for reverse DNS, returned similar data to dnsrecon and led me nowhere.
8. ip2location - Helped me determine that this server was operated by Evry
9. Maltego - Keeping data structured in a graph
10. ~~Pipl.com~~ - Attempting to find information about employees based on name
11. theharvester - Gathers information via a set of search engines + reverse DNS lookup
12. Wappalyzer - Identifying server information from regex'ing returned data
13. whois - Query registrar for information about domain ownership
14. ~~V3n0md0rk3r~~ - Fork of smartd0rk3r for querying search engines with over 10000 Google d0rks for a specific domain to locate potential SQL-injections. Nothing found (IIS URL rewrite module activated on server)

2 Review: Rogue Code - A Jeff Aiken Novel

So, as the assignment was to read the book **Rogue Code: A Jeff Aiken Novel**, I admit to being rather hesitant to reading this book. However, after getting into it I quickly admitted to having judged the book too fast, and after I had begun I couldn't put the book down, it was just too darn good and I wish to express my biggest **thank you** to whoever made the decision that we should read this book: It's something I have missed so badly, to read a book with an actual storyline rather than just dry facts, but still keeps on topic. I really appreciate it!

As for the book, though, it tells the story of Jeff Aiken, a former CIA officer who now runs his own security consultant company, RedZoya. A group of attackers perform reconnaissance against a Wall Street trading company, mapping their network and digging through documents. They manage to get a pretty good picture of the structure of the network, and by chance stumble upon an unpatched vulnerability in one of the jump servers, and manage to get in.

RedZoya is hired to perform a penetration test after a "harmless" bot is found on one of the jump servers, and stumble upon a rootkit which Jeff begins reverse engineering, not knowing it's part of an ongoing operation by the Brazilian organized crime group Nosso Lugar, lead by *chefe* Victor Bandeira and the operation being spearheaded by his son, Pedro, as part of his (partly unknowing) training. Pedro doesn't want to be a criminal, but wants to truly gets to know his father, whom he has been upset with since he divorced his mother - something seen as dishonorable in Brazil. Taking his mothers advice and putting the past behind him, he reconciles with his father and takes his suggestion of running his own company, Casa de Férias, which is also a shell company for infiltrating the New York Stock Exchange for a long time while planting code and preparing for the big heist against Toptical's IPO, Carnaval.

Nosso Lugar's new leadership had seen the potential in computers early, and had hired Abílio Ramos as he had seen potential in him early. Ramos had helped Bandeira take control of their large online gambling sites, and Abílio had led *Distributed Denial of Service* attacks against other large gambling sites with the *botnets* they had built by infecting unsuspected users through *drive-by downloads*, click-baiting (presumed from the mention of being *penetrated by vulnerabilities in their web browser*), or *trojans* sent as email attachments. Using these botnets, they could blackmail others into submission and eliminate competitors. These botnets also placed spyware on the target computers, allowing Nosso Lugar to loot bank accounts, and malware to distribute spam – I believe Russinovich drew inspiration from the rise of exploit kits such as Zeus and BlackHole here. Russinovich uses **strong** real-world correlation and high technical correctness to make his story believable, rather than leaving it entirely to believable human interaction or depth in personality, which I feel is a great (but typical) techie-approach, and placing subtle undertones with criticism of certain (often 3 letter) agencies, by briefly mentioning PRISM and the relentless prosecution of Aaron Swartz. Toptical seems to correspond to rising competitors to Facebook focusing on privacy and ethics (Ello, di-

aspora*), and Brazil seems to be a substitution of Russia (perhaps to avoid yet another work making Russia the antagonist, as is again becoming increasingly common in this digital era cold war).

This is where the book gets interesting. Nosso Lugar have infiltrated the NYSE over half a decade by assigning their best technically, albeit not socially (considered he hired fellow NYSE sociopath colleague Richard Iyers as his partner), skilled employee Abílio undercover under direct management of the younger Bandeira, Pedro, to work with the digital infrastructure. Having worked for years, Abílio has gained more and more access to the infrastructure as a trusted employee under the name of Marc Campos, and has been able to infect the servers with Nosso Lugar's obfuscated code.

Jeff & Frank, having been hired to perform a penetration test to see how this suspicious bot could have got into the NYSE jump servers, Jeff employ one of his own tools to create a map of the systems in the network from an unprivileged system, just like an attacker from the outside would've done. The book often mentions tools generically without calling them by name, and most tools are custom written by Jeff (Aikens being the Tony Stark of Information Security) - some being his private ones, and some publicly distributed at presentations.

As Jeff developed an overview of the network, Frank went through the NYSE's intranet directory looking for public documents he could scan for information related to the jump servers. Just like the attackers who placed the bot there, he found some documents for the Universal Trading Platform containing names and accounts for the team deploying trading software to the engines in New Jersey.

Jeff & Frank's reconnaissance of the sites, servers, antiviruses, user account and their roles had led them to realize the exchange's biggest customers had unnecessarily high permissions, and that the network was split in two – one unprivileged untrusted zone, and one trusted zone not facing the internet where the action took place, with the two zones being connected by jump servers.

Jeff's company, RedZoya, purchased a lot of exploits from security research firm FirstReact, which was also the firm that had reported vulnerabilities, sold exploits for these vulnerabilities as soon as patches were released, further increasing the need to patch. One of these vulnerabilities had been left unpatched on the jump server, and as Jeff had access to the exploit, they were able to get a foothold on the jump server.

These zones were accessed by logging into the jump server through an account with specific access to that zone; Jeff & Frank thus proceeded to dig through access-logs for the specific jump server, to locate a frequent accessor. Having located this user, they used a Pass-the-Hash attack (considering this book was released in May 2014 and Russinovich held a presentation about PtH at RSA in February, I believe this was written as his personal creative expression of having worked a lot with this type of attack) – using the cached hash of the users password to tag along with the user into the system. Policy was that users should never use privileged accounts to log in through these servers as it's a considerably high security risk, but as is often the case with policy, laziness wins in the users eyes and as such they were able to freeload into the system with higher-than-should've-been privileges.

This granted them administrative privileges on the *insecure* network, but after performing reconnaissance of the administrator's computer they now had access to, and going through his documents and email, they could identify the log-in process to the jump server as two-factor requiring a physical device. Thus, they set up an alarm and waited for the administrator to make a connection to the jump server, and piggyback into it.

After gaining access, they placed two backdoors (one as a back-up), and established a connection to the jump server from which they could further gain access to the trusted zone. Jeff began reverse engineering the obfuscated file hidden through the rootkit his private tool, *Rotorooter*, had identified, but being driven by his own pride to show something amazing all at once, instead of showing progress step-by-step, he made a major mistake in failing to report the extent of his progress to their employer, Bill Stenton at the NYSE. This, as one may expect, lead to their major downfall later on, creating a lack of trust between Jeff & Frank, and their employer.

Nosso Lugar begin to realize Jeff is onto them, and they take two different approaches to the problem. Richard Iyers turns out to be a loose cannon bordering on sociopath, and becomes seemingly uncontrolled - while the Brazilian infiltrator, Campos (Abílio), as a veteran in Nosso Lugar and with previous experience, gets pushed to take extreme measures by framing Jeff by planting code that makes obvious trade thefts regardless of the time of day doomed to be detected by the automated security system rapidly funneling the money straight into an account opened in Jeff's name using information retrieved by social engineering, the loose cannon (Iyers) is not as tactical and much

more violently inclined, and instead makes an attempt at murdering Jeff.

At the same time, the people at the largest fictional social network site Toptical is about to go public on the New York Stock Exchange, and although they have internal conflicts of whether to be acquired by one of the tech giants like Google, Microsoft, etc, or go public with the risk of their stocks being valued too high and risking High Frequency Traders (trading bots) messing the prices up, they have already settled on the decision to go public.

This is where the book starts getting truly intriguing, and following the dramaturgical curve almost religiously, having performed the penetration test at around 25% and introduced all major players but one, at around 33% the U.S Securities and Exchange Comission, SEC, gets involved and is determined to put Jeff behind bars, shifting focus from attack and penetration testing, to Jeff & Frank trying to clear his name and Nosso Lugar trying to pull off their big heist as Toptical goes public, skimming money off of specific stocks at specific prices, as their rogue code places itself in front of the line and getting the best prices first. Now the snowball that is the plot has been put in motion, culminating in Jeff & Frank being on the run and Jeff's ex-wife and love-of-his-life Daryl (I wonder who Russinovich let get away?) is called in by Frank to help them clear their names.

As Frank & Jeff flee the country under false identities (because breaking any law is okay by law enforcement as long as you're a good guy) to take the bait given to them by Nosso Lugar in the form of geo-embedded images containing coordinates and a threat, Daryl uses her insanely attractive looks to infiltrate the NYSE through social engineering - because apparently the NYSE has absolutely no useful safeguards against anybody determined to get in - and eventually identifies Campos as the culprit of the operation. She borrows his phone without permission and identifies a Bluetooth-vulnerability she knows she has been able to exploit before to unlock it, and copies all of his personal data off the phone (something which could have been prevented by updating firmware and Android), allowing her to provide Jeff & Frank with more detailed information - but as the only woman of any importance in the book she ends up getting raped by the wildcard, Iyers. Luckily she's also a martial arts ninja (or at least trained in self-defense) and killed the rapist, with the local law enforcement unable to identify any suspects.

Rogue Code very clearly defines the web of protagonists and antagonist, starting with

Nosso Lugar vs NYSE, and then pitted against Jeff & Frank; who in turn are later pitted against the SEC; Mitri Growth trying to profit off of Toptical. These different groups at first seem like different strings, but as the plot thickens they become more and more interweaved, the plot really thickening at around 50% and culminating with a shoot-out at 85-90%, after which the decline quickly follows with Jeff and Frank being cleared with the help from Jeff's ex-wife Daryl (to throw in a love story to carry all of this), Nosso Lugar's leader family being eliminated, the NYSE being saved *just in time*, Mitri Growth profiting slightly while Toptical fails badly (presumably to prove the point of the problems with Wall Street, where those who contribute nothing may profit off of those who do) and Jeff and Daryl being reunited.

An interesting point brought up by Russinovich in this Jeff Aiken novel is the High Frequency Traders. I was previously aware of automated trade bots, it comes as no surprise, but how they worked and to what extent they affect the global financial market was something I hadn't considered before. The fact that they can manipulate the market to almost fix prices by performing trades at insanely high speeds and profiting both when prices go up and down, and the volatility this brings, and the effect latency has on this, I never really stopped to consider; which brings me to further question reasons for the FCC wanting to allow ISPs to charge extra for high speed and prioritized internet traffic.

As HFTs are automated and require no interaction, they could (can?) issue non-executable orders in batches, intended to detect early trends (or clog up the exchanges with noise). Eventually, you have machines that contribute nothing to the market but constantly take their cut from trades, their only contribution being a new algorithm - money disappearing into somebody else's pockets for nothing.