



HSR
HOCHSCHULE FÜR TECHNIK
RAPPERSWIL

COMPUTER SCIENCE

STUDYTHESIS

Readiness for Tailored Attacks and Lateral Movement Detection

Authors:

Claudio MATTES
claudio.mattes@hsr.ch

Lukas KELLENBERGER
lukas.kellenberger@hsr.ch

Supervisor:

Cyrill BRUNSCHWILER
Hochschule für Technik Rapperswil
cyrill.brunschwiler@hsr.ch

DEPARTEMENT COMPUTER SCIENCES
HSR UNIVERSITY OF APPLIED SCIENCES RAPPERSWIL
CH-8640 RAPPERSWIL, SWITZERLAND

October 15, 2018

Abstract

Management Summary

Initial Situation

Procedure

Results

Outlook

Task Definition

Readiness for Tailored Attacks and Lateral Movement Detection

Aufgabenstellung SA Herbst 2018

Datum: September 28., 2018
Author: Cyrill Brunswiler, Compass Security Schweiz AG
Classification: INTERNAL

Table of Contents

| | | |
|-----------|----------------------------------|----------|
| 1 | EINFÜHRUNG | 3 |
| 2 | AUFGABE | 3 |
| 2.1 | Abgrenzung..... | 3 |
| 2.2 | Tätigkeiten | 3 |
| 3 | VORGEHEN | 3 |
| 4 | ANFORDERUNGEN..... | 3 |
| 4.1.1 | Technologien | 4 |
| 5 | INFRASTRUKTUR | 4 |
| 6 | ERWARTETE RESULTATE | 4 |
| 6.1 | In elektronischer Form: | 4 |
| 6.2 | Auf Papier: | 4 |
| 7 | TERMINE..... | 4 |
| 7.1 | Start/Ende | 4 |
| 7.2 | Zeitplan und Meilensteine | 4 |
| 8 | BETREUUNG | 5 |
| 8.1 | Kontakt..... | 5 |
| 9 | REFERENZEN..... | 5 |
| 10 | UNTERSCHRIFTEN | 5 |

1 Einführung

Es werden vermehrt Cyberangriffe publik, wo Schadcode im Einsatz ist, welcher sich nicht nur auf einem infizierten System niederlässt, sondern weitere Systeme im Netz befällt. Das Ziel oder Resultat ist dabei oft die komplette Infiltrierung einer Organisation. In der Analyse solcher Fälle sind Information und Zeit ein Schlüssel zum Erfolg. Folglich ist die Bereitschaft "Readiness" für ein solches Ereignis ein entscheidender Faktor.

2 Aufgabe

Ziel dieser Arbeit ist es, ein Tool zu erstellen, welches die Bewertung der eigene Readiness erlaubt aber auch im Analysefall eine Unterstützung bietet. Readiness betrifft viele Aspekte und einfache Dinge wie korrekte Zeitstempel in Logs, deren Vollständigkeit oder die Bereitstellung von Backups. In der konkreten Aufgabenstellung soll die Readiness-Analyse primär für Windows-Infrastrukturen anhand von Logs und spezifischen Events erfolgen. Unter anderem soll auf den neusten Publikationen des japanischen Computer Emergency Response Teams (JPCERT/CC) und der öffentlichen Datenbank der MITRE Corporation, dem Adversarial Tactics, Techniques, and Common Knowledge (ATT&CK™) Wissenspool, basiert werden. Das JPCERT und MITRE haben dabei die Werkzeuge und das generelle Vorgehen von Angreifern analysiert und geben Hinweise, welche Events auf eine mögliche Verseuchung hinweisen.

2.1 Abgrenzung

Es geht nicht darum neue Angriffsvektoren zu finden.

2.2 Tätigkeiten

- Projektmanagement und Dokumentation
- Einarbeitung in Incident Handling und Forensik
- Einarbeitung in Angriffstechniken und Werkzeuge
- Einarbeitung in Abwehrtechniken und Härtung von Systemen
- Studium öffentlicher Quellen und verfügbaren Tools
- Umsetzung eines Analyzers gemäss Anforderungen basierend auf etablierten Frameworks

3 Vorgehen

Im Rahmen der allgemeinen Richtlinien zur Durchführung von Studien- und Bachelorarbeiten gemäss eigenem Projektmanagementplan. Dieser Projektmanagementplan ist als Erstes zu erstellen und enthält insbesondere:

- Die Beschreibung des dem Projektcharakter angepassten Vorgehensmodells.
- Eine erste Aufteilung der Aufgabe in gemeinsam und einzeln zu bearbeitende Teile unter Berücksichtigung der vorgegebenen Teilaspekte. Die genaue Aufteilung muss spätestens nach der Technologiestudie (Elaboration) erfolgen.
- Den Projektplan (Zeitplan) und die Meilensteine.

4 Anforderungen

Es geht primär darum einen Analyzer zu erstellen um die "Readiness for Tailored Attacks and Lateral Movement Detection" beurteilen zu können. Idealerweise kann dieses Tool von einem IT Administrator ohne spezielle Kenntnisse und grossartige Installationsprozedur ausgeführt werden.

Schematisch aber nicht bindend werden folgende Schritte auszuführen sein

- Definition der Requirements für einen neuen/verbesserten Analyzer
- Design und Analyse basierend auf den Vorgaben
- Vorschläge für die Umsetzung oder Verbesserung eines
 - Readiness Analyzers
 - Readiness Optimizers
 - Compromise Analyzers
- Implementation der Funktionalität und Erstellung eines Benutzerhandbuch
- Erweiterung der Analyzer um neue Erkenntnisse, Werkzeuge und Indicators
- Dokumentation der Software und Skripte

4.1.1 Technologien

- Windows Workstations, Windows Server, Windows Security generell
- Windows Event Logs, Security und Audit Logs
- Windows On-Board Tools, Sysinternals Toolkit
- Active Directory Service (AD) Services
- Group Policy Objects (GPO)
- PowerShell, .NET, Python, Windows Batch

5 Infrastruktur

Die Arbeiten werden auf den Rechnern der Studenten durchgeführt. Zusätzlich benötigte Software oder Hardware wird bei Bedarf und nach Rücksprache mit Compass Security zur Verfügung gestellt.

6 Erwartete Resultate

6.1 In elektronischer Form:

- lauffähiges Toolkit und kompletter Source Code
- komplette Software Dokumentation (Use Cases, Klassenmodell, Sequenzdiagramme usw. in UML)
- komplette Use Cases und Erfolgs-Szenarien resp. Musterlösungen
- alle Dokumente und Protokolle (vorzugsweise in englischer Sprache)

6.2 Auf Papier:

Gemäss der Anleitung der HSR: \\hsr.ch\root\alg\skripte\Informatik\Fachbereich\Studienarbeit_Informatik

Es muss aus den abgegebenen Dokumenten klar hervorgehen, wer für welchen Teil der Arbeit und der Dokumentation verantwortlich war (detaillierte Zeiterfassung).

7 Termine

7.1 Start/Ende

- Termine gemäss \\hsr.ch\root\alg\skripte\Informatik\Fachbereich\Studienarbeit_Informatik\SAI\Termine

| Datum | Task |
|------------|--|
| 17.09.2018 | Beginn der Arbeit, Ausgabe der Aufgabenstellung durch den Betreuer. |
| 18.12.2018 | <p>Erfassung des Abstracts im Online-Tool https://abstract.hsr.ch/ Die Studierenden geben den Abstract für die Diplomarbeitbroschüre zur Kontrolle an ihren Betreuer/Examinator frei.</p> <p>Der Betreuer/Examinator gibt das Dokument mit dem korrekten und vollständigen Abstract zur Weiterverarbeitung an das Studiengangsekretariat frei.</p> <p>Vorlagen sowie eine ausführliche Anleitung betreffend Dokumentation stehen auf dem Skripteserver zur Verfügung.</p> |
| 21.12.2018 | Hochladen aller verlangten Dokumente auf archiv-i.hsr.ch Abgabe des Berichts an den Betreuer bis 12.00 Uhr |

7.2 Zeitplan und Meilensteine

Zeitplan und Meilensteine für das Projekt sind von den Studenten selber zu erarbeiten und zusammen mit dem Projektmanagementplan abzuliefern. Die Meilensteine sind bindend. Der erste Meilenstein ist vorgegeben. Mit den Betreuern werden regelmässige Sitzungen zur Fortschrittskontrolle durchgeführt.

8 Betreuung

Die Arbeiten werden durch Cyrill Brunschwiler betreut. Der Gegenleser ist noch nicht bestimmt.

8.1 Kontakt

Cyrill Brunschwiler, Managing Director, Compass Security Schweiz AG
Weststrasse 50, 8003 Zürich, Switzerland
Werkstrasse 20, 8645 Jona, Switzerland

+41 55 214 41 73

cyrill.brunschwiler@compass-security.com

cyrill.brunschwiler@hsr.ch

<https://fb.com/compass-security.com/inbox/hUGXMr2EeZ2V7b>

9 Referenzen

- JPCERT/CC Detecting Lateral Movement through Tracking Event Logs https://www.jpcert.or.jp/english/pub/sr/20170612ac-ir_research_en.pdf
- JPCERT/CC Detecting Lateral Movement through Tracking Event Logs v2 https://www.jpcert.or.jp/english/pub/sr/Detecting%20Lateral%20Movement%20through%20Tracking%20Event%20Logs_version2.pdf
- JPCERT/CC Detecting Lateral Movement in APTs, <https://www.first.org/resources/papers/conf2016/FIRST-2016-105.pdf>
- JPCERT/CC Online Results Sheet, <https://jpcertcc.github.io/ToolAnalysisResultSheet/>
- JPCERT/CC Logon Tracer, <https://github.com/JPCERTCC/LogonTracer>
- CERT-EU Security Whitepaper 17-002, http://cert.europa.eu/static/WhitePapers/CERT-EU_SWP_17-002_Lateral_Movements.pdf
- NSA Spotting the Adversary, <https://www.iad.gov/iad/library/ia-guidance/security-configuration/applications/spotting-the-adversary-with-windows-event-log-monitoring.cfm>
- MS (Sysinternals) Sysmon <https://docs.microsoft.com/de-ch/sysinternals/downloads/sysmon>
- MS Logparser <http://www.microsoft.com/en-us/download/details.aspx?id=24659>
- MS Windows Defender ATP Advanced Hunting <https://github.com/Microsoft/WindowsDefenderATP-Hunting-Queries>
- MS Poorman Monitoring <https://blogs.technet.microsoft.com/jepayne/2015/11/23/monitoring-what-matters-windows-event-forwarding-for-everyone-even-if-you-already-have-a-siem/>
- MITRE ATT&CK Adversarial Tactics, Techniques & Common Knowledge <https://attack.mitre.org/>
- The CALDERA automated adversary emulation system <https://github.com/mitre/caldera>
- The APT Simulator Windows Batch <https://github.com/NextronSystems/APTSimulator>
- Infection Monkey - An automated pentest tool <https://github.com/guardicore/monkey>
- Flightsim - A utility to generate malicious network traffic and evaluate controls <https://github.com/alphasoc/flightsim>

10 Unterschriften

Jona, 28. September 2018



Cyrill Brunschwiler



Claudio Mattes



Lukas Kellenberger

Contents

| | |
|--|------------|
| Abstract | I |
| Management Summary | II |
| Initial Situation | II |
| Procedure | II |
| Results | II |
| Outlook | II |
| Task Definition | III |
| Contents | X |
| I Technical Report | XI |
| 1 Introduction and Overview | 1 |
| 2 Analysis | 2 |
| 2.1 BloodHound / SharpHound | 2 |
| 2.1.1 Description | 2 |
| 2.1.2 Difficulties | 2 |
| 2.1.3 Conclusion | 2 |
| 2.2 WEFFLES | 2 |
| 2.3 Microsoft Security Compliance Toolkit | 3 |
| 2.3.1 Description | 3 |
| 2.3.2 Difficulties | 3 |
| 2.3.3 Conclusion | 3 |
| 2.4 Microsoft Monitoring Active Directory for Signs of Compromise | 3 |
| 2.4.1 Description | 3 |
| 2.4.2 Conclusion | 4 |
| 2.5 MITRE ATT&CK | 4 |
| 2.6 JPCERT/CC - Detecting Lateral Movement through Tracking Event Logs . . . | 4 |
| 2.7 JPCERT/CC - Detecting Lateral Movement in APTs | 4 |
| 2.8 LogonTracer | 4 |
| 2.8.1 Description | 4 |
| 2.8.2 Difficulties | 5 |
| 2.8.3 Conclusion | 6 |
| 2.9 Test environment | 6 |
| 2.9.1 Users | 7 |
| 3 System Architecture | 8 |
| 3.1 Use Cases | 8 |

| | | | |
|------------------------|-------|--|-------------|
| | 3.1.1 | UC01 - Read Event Logs | 8 |
| | 3.1.2 | UC02 - Analyse Event Logs | 8 |
| | 3.1.3 | UC03 - Read Audit Policies | 9 |
| | 3.1.4 | UC04 - Analyse Audit Policies | 9 |
| | 3.1.5 | UC05 - Display missing or wrong system configuration | 9 |
| | 3.2 | Non Functional Requirements | 10 |
| | 3.3 | Technologies | 10 |
| 4 | | Results | 11 |
| 5 | | Conclusion | 12 |
| Glossary | | | VI |
| List of Figures | | | VII |
| List of Tables | | | VIII |

Part I

Technical Report

1 Introduction and Overview

As described in the introduction of the task definition the key for a successful analysis in case of an advanced persistence threat (APT) or lateral movement in a network, it is fundamental to have solid event logging of all systems participating in the network.

Shusei Tomonaga at the Japan Computer Emergency Response Team Coordination Center (JPCERT/CC) has shown with the study "Detecting Lateral Movement through Tracking Event Logs" [1] how important it is to configure solid event logging to analyze attacks. JPCERT/CC found in their study that APT and lateral movements could be detected with the correct settings in the audit policy and with the help of Sysmon 37 of 44 attacks.

2 Analysis

This chapter describes the first step of this project, the research of published technical reports and tools which are considered interesting for this project.

2.1 BloodHound / SharpHound

2.1.1 Description

BloodHound describes itself on its wiki page on GitHub as follows:

"BloodHound is a single page Javascript web application, built on top of Linkurious, compiled with Electron, with a Neo4j database fed by a PowerShell/C# ingestor.

BloodHound uses graph theory to reveal the hidden and often unintended relationships within an Active Directory environment. Attacks can use BloodHound to easily identify highly complex attack paths that would otherwise be impossible to quickly identify. Defenders can use BloodHound to identify and eliminate those same attack paths. Both blue and red teams can use BloodHound to easily gain a deeper understanding of privilege relationships in an Active Directory environment." [2]

2.1.2 Difficulties

BloodHound was tested in the test environment describes later in this chapter. Both, the C# and Python ingestors were successfully installed and tested. The only problem which occurred was that the Python-ingestor does not yet run on the latest Python release. One must have a Python 2.7.x version installed to run the scripts successfully.

2.1.3 Conclusion

The, for our project, most interesting part about BloodHound is the way they retrieve their data. Due to the decision that the application, in a first step, only reads the data of the local computer and not the whole domain, BloodHound will only be important in a later part of the project.

2.2 WEFFLES

WEFFLES (Windows Event Logging Forensic Logging Enhancement Services) is a Threat Hunting/Incident Response Console with Windows Event Forwarding and PowerBI, coded and published by Microsoft-Security-Employee Jessica Payne. It is build to help setting up the Windows Event Forwarding, so that all the collected logs of a system are stored on one centralized server, and afterwards to analyse the collected data. Jessica Payne wrote an installation instruction on the Microsoft TechNet blog <https://blogs.technet.microsoft.com/jepayne/2017/12/08/weffles/>. Once the data is collected one could simply import the generated weffles.csv file into Excel and start filtering the logs to gain the needed. Jessica Payne recommends to use PowerBI, a business analytics tool designed by Microsoft. In her published blog she also gives a short introduction on what to look out for, which event ids are important and other useful tips and tricks for detecting suspicious activities in the network.

WEFFLES will not be the product on which this project is based, but could become an important point of reference.

2.3 Microsoft Security Compliance Toolkit

2.3.1 Description

The Microsoft Security Compliance Toolkit (SCT) [3] allows security administrators to analyze their configured enterprise Group Policy Objects (GPO) in comparison to the Microsoft-recommended GPO baselines. The toolkit is handed with several baseline GPO's for different versions of Microsoft Windows Client and Servers:

- Windows 10 security baselines
 - Windows 10 Version 1803 (April 2018 Update)
 - Windows 10 Version 1709 (Fall Creators Update)
 - Windows 10 Version 1703 (Creators Update)
 - Windows 10 Version 1607 (Anniversary Update)
 - Windows 10 Version 1511 (November Update)
 - Windows 10 Version 1507
- Windows Server security baselines
 - Windows Server 2016
 - Windows Server 2012 R2
- Microsoft Office security baseline
 - Office 2016

2.3.2 Difficulties

The toolkit is very simple and could be understood and used without any difficulties. The handling is very intuitive and does not require much training. Please note, however, that the toolkit cannot be used with Windows 10 Home, since active directory support is not provided with this version.

2.3.3 Conclusion

This toolkit can be used for a very baseline GPO in enterprise environment. With the handed baselines it is easy to compare the configured GPO and to see the readiness of the enterprise GPO. The toolkit gives also the ability to compare different local GPO's installed on different Clients or Servers to check their consistency. In addition the handed baselines can be used for building new GPO's.

This toolkit is very interesting, but cannot be used to build on it. The reason for this is that the source code of the complete toolkit is not available. However, it can be used as additional help for checking the readiness of an enterprise environment.

2.4 Microsoft Monitoring Active Directory for Signs of Compromise

2.4.1 Description

This article [4] is about configuration of an solid event log monitoring for Microsoft servers. The article gives a quiet well overview about the audit policy in Microsoft systems and what each policy stands for. The article gives information about the most important audit policies and how noisy (if a lot of data is produced by them) they are. This study does not go into the details of the audit policies in

detail. Furthermore the article describes how the policies can be read with powershell.

To this article Microsoft compiles in Appendix L [5] all important event ids which are necessary for a successful detection of APTs and lateral movements.

2.4.2 Conclusion

Due to the fact that audit policies are an important setting for solid event logging, this article and appendix L will be a central part of the toolkit to be built. As a next step and part of this study these event ids have to be correlated with the event ids found in the JPCERT/CC's study "Detecting Lateral Movement through Tracking Event Logs" [1] to make a clear statement which event ids have to be logged.

2.5 MITRE ATT&CK

2.6 JPCERT/CC - Detecting Lateral Movement through Tracking Event Logs

This is a document the Japan Computer Emergency Response Team Coordination Center, or short JPCERT/CC, has published in the year 2017. It describes how, in their experience, attackers proceed with lateral movement. In a very detailed 81-page report they describe step by step the procedure, the tools used and what is most interesting for the project, the logs generated while doing so.

This report is going to have a big impact on this project, it shows which logs have to be read in any case.

2.7 JPCERT/CC - Detecting Lateral Movement in APTs

This document is from a presentation by Shingo Abe, a JPCERT/CC employee. In it he describes how to find system intruders more effectively using Windows Event Logs. The collected data is used to better detect inconsistencies, such as when an administrator logs on to another machine or when an administrator logs on suspiciously often.

This presentation contains interesting information which could be built into the project at a later point.

2.8 LogonTracer

2.8.1 Description

JPCERT/CCs LogonTracer is a tool built to investigate malicious logons on a system based on the research described in section 2.7 JPCERT/CC - Detecting Lateral Movement in APTs. The tool links hostnames or IP addresses with the *"account name found in logon-related events and displays it as a graph"*. [6] The following event ids are checked with the tool:

- 4624:Successful logon
- 4625:Logon failure
- 4768:Kerberos Authentication(TGT Req.)
- 4769:Kerberos Service Ticket (ST Req.)
- 4776:NTLM Authentication
- 4672:Assign special privileges

The following figure shows a sample graph from the test environment:

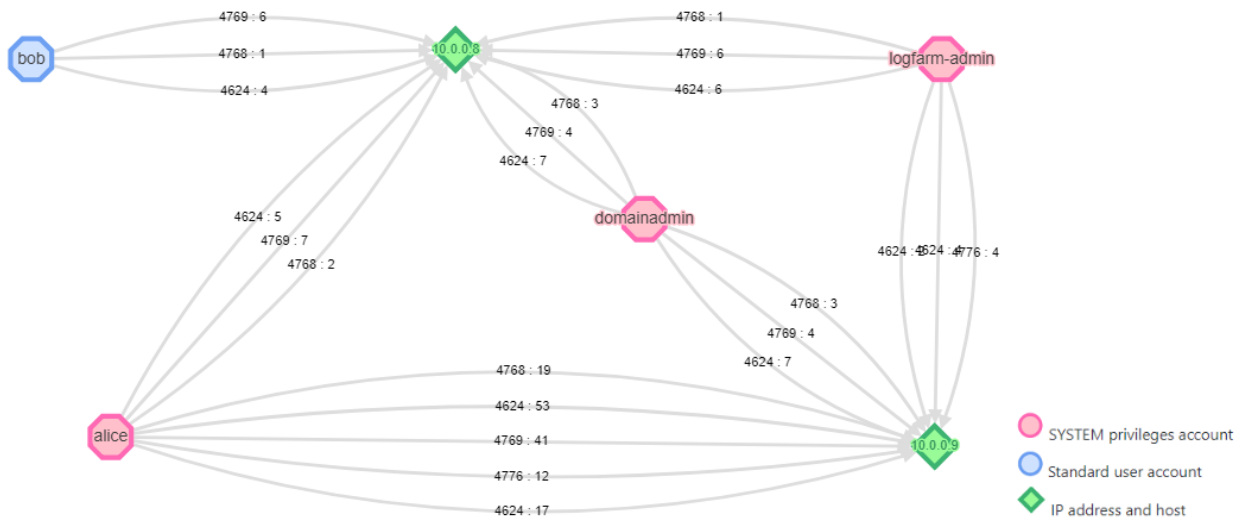


Figure 1.1: LogonTracer: Sample Graph from Test Environment

To use the LogonTracer, there is only a .evtx-File (Windows XML Event Log - export of Windows event logs) needed to be uploaded. At best the export of the security event log of the domain controller to get as much information of the network as possible. With the built in analysis of logins, by using machine learning models and statistical analysis, LogonTracer is able to provide a ranking of the most malicious users which tried to log in.

In addition, LogonTracer provides a timeline for all or selected users to see when each user logged in. The timeline can also be displayed as a graph with the LogonTracer, allowing anomalies to be detected more quickly.

The test environment showed that this graph can quickly become confusing - especially in a larger corporate environment. Although only a small environment as described in the section was used, it turned out that various users wanted to log on to the virtual machines. The reason for this is that the test environment is accessible via public IP addresses in the cloud.

Nevertheless, with meaningful filters the search can be restricted and the graph can be used efficiently, as shown in the figure 1.1 LogonTracer: Sample Graph from Test Environment

2.8.2 Difficulties

During the test phase of LogonTracer some difficulties were detected. It's pretty easy to get the docker container, but starting LogonTracer is a bit of a challenge. JPCERT/CC gives the following instructions for starting the docker container:

Listing 1.1: LogonTracer: given docker run command

```

1 $ docker run --detach \
2   --publish=7474:7474 --publish=7687:7687 --publish=8080:8080 \
3   -e LTHOSTNAME=[IP_Address] jpcertcc/docker-logontracer

```

The Problem here was that the parameter [IP_Address] was not described well. If the command `docker ps` was executed it always showed the following PORTS:

Listing 1.2: LogonTracer: docker ps (PORTS)

```

1  PORTS
2  0.0.0.0:7474->7474/tcp, 0.0.0.0:7687->7687/tcp, 7473/tcp, 0.0.0.0:8080->8080/tcp

```

After a lot of investigation and further tests it turned out that under **PORTS** the ports resp. ip addresses of the container can be bound to the host. But these are not relevant for the LogonTracer, because it provides a web application under the defined parameter **[IP_Address]** and after enough patience it can be reached via **localhost:8080**. If this parameter was set to **127.0.0.1**, the database containing the imported .evtx file could not be accessed. Thus the graph was never displayed. The parameter **[IP_Address]** set to **localhost** also solved this problem.

Listing 1.3: LogonTracer: recommended docker run command

```

1  $ docker run --detach \
2  --publish=7474:7474 --publish=7687:7687 --publish=8080:8080 \
3  -e LTHOSTNAME=localhost jpcertcc/docker-logontracer

```

2.8.3 Conclusion

The LogonTracer is unique in its form and should not be underestimated for the detection of lateral movements. This is because user access to the components available in the network can be easily visualized in graphical form.

However, the LogonTracer is not suitable for detection readiness and cannot be used to build on it. However, approaches for reading the event log for further work could be used. And for a further detection of lateral movements this tool is extremely interesting.

2.9 Test environment

A virtual network was set up on Azure-Cloud as a test environment. The test network was set up in the cloud so that the development team can access the network regardless of its location. The test network consists of a Windows server and two Windows clients. Active Directory service was configured on the server to manage the client computer. The following operating systems were installed in this test network:

Server:

- Windows Server 2016

Clients:

- Windows 10 Pro, Version 1709

The network is structured as followed:

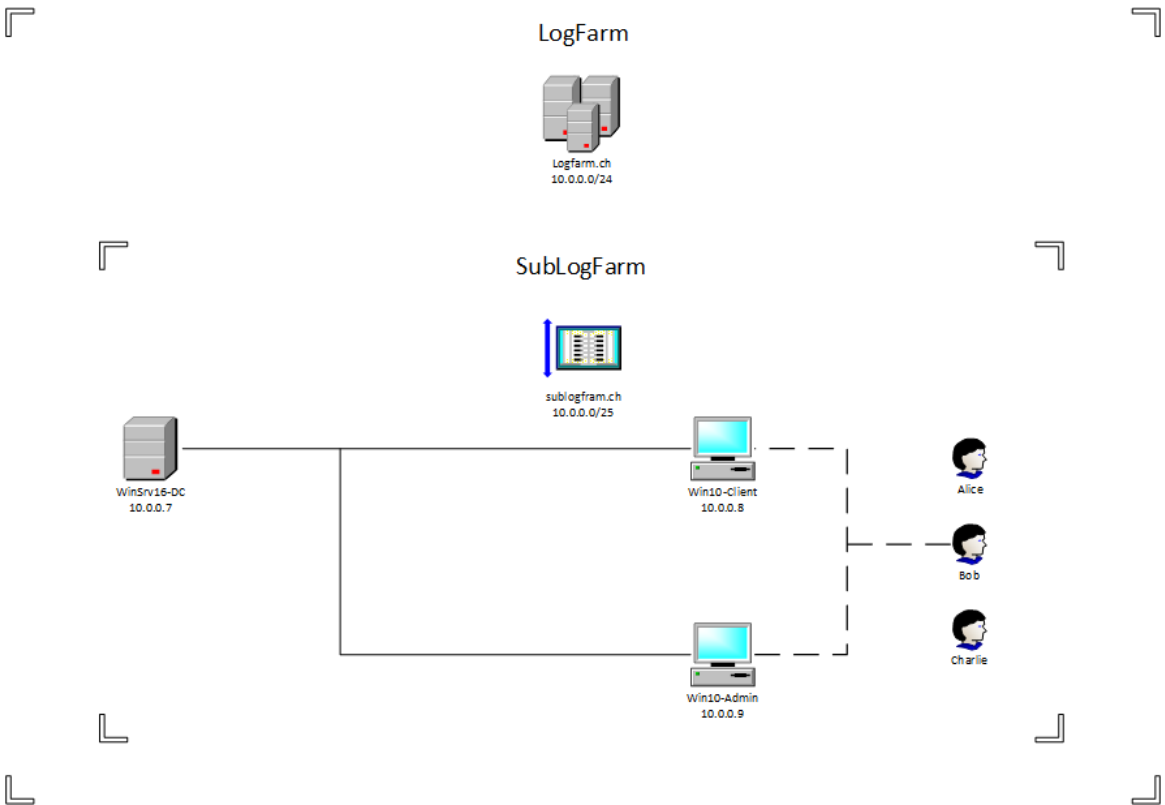


Figure 1.2: Test Environment

2.9.1 Users

Three different users were configured:

| Name | Permissions |
|---------|---------------|
| alice | administrator |
| bob | user |
| charlie | user |

Table 1.1: Test Environment Users

3 System Architecture

In this section the following main question is answered:

"How would a system architecture look like to fulfill the described problem domain?"

This includes the coverage of use cases, non-functional requirements, technologies used ...

3.1 Use Cases

A visual representation of the use cases with a use case diagram was deliberately omitted, because there is only one actor involved - the security advisor. The actor is not specifically mentioned in the use cases every time, because it is always the same.

3.1.1 UC01 - Read Event Logs

Description

Event logs are read from the running system and saved in a temporary file.

Precondition

The system is running and must have valid event logs.

Main Success Scenario

1. Read the specified event logs from the local system
2. Save the needed information from the event logs in a temporary file for analysis purposes.

3.1.2 UC02 - Analyse Event Logs

Description

The implemented logic analyzes, by defined event ids, which event occurred or is missing and creates a list of events that did not occurred or are not logged yet.

Precondition

UC01 is fulfilled: the temporary file is available.

Main Success Scenario

1. The temporary file can be read
2. The list with the defined event ids is available
3. Create a list of events which occurred and which are missing

3.1.3 UC03 - Read Audit Policies

Description

The specified domain audit policies are read and saved in a temporary file.

Precondition

TODO: check for admin rights needed

Main Success Scenario

1. Read the specified domain audit policies from the system
2. Save the needed information from the audit policies in a temporary file for analysis purposes.

3.1.4 UC04 - Analyse Audit Policies

Description

Based on the list created in UC02, the implemented logic analyzes whether the missing events did not occur or never occurred due to the incorrect configuration. If this is not the case, the system checks whether the events are logged at all using the audit policy.

Precondition

UC02 and UC03 are fulfilled: the temporary files are available.

Main Success Scenario

1. The temporary files can be read
2. The list with the defined audit policies is available
3. Creates a list of events where the logging is not configured

3.1.5 UC05 - Display missing or wrong system configuration

Description

Based on the list created in UC04 the user gets an overview of missing configurations which would improve the readiness of the system for a good attack detection.

Precondition

The list from UC04 is available.

Main Success Scenario

1. Displays a visual output of missing or wrong system configuration

3.2 Non Functional Requirements

| NFR-No. | Description |
|---------|--|
| NRF01 | The Toolkit must remain the system in the status quo. More specific the system must not change or remove any existing entry in the eventlog, registry as well as in the execution history. |
| NFR02 | The user must not notice any performance degradation from the system when using the Toolkit. |
| NFR03 | The Toolkit must be portable with no installation procedure before use. |
| NFR04 | The target version of the system for the Toolkit to run must be Microsoft Windows 10 Professional or Microsoft Server 2016. |

Table 1.2: Non Functional Requirements

3.3 Technologies

4 Results

5 Conclusion

Glossary

List of Figures

| | | |
|-----|---|---|
| 1.1 | LogonTracer: Sample Graph from Test Environment | 5 |
| 1.2 | Test Environment | 7 |

List of Tables

| | | |
|-----|---------------------------------------|----|
| 1.1 | Test Environment Users | 7 |
| 1.2 | Non Functional Requirements | 10 |

Bibliography

- [1] JPCERT/CC. Detecting Lateral Movement through Tracking Event Logs. , 2017.
- [2] JPCERT/CC. BloodHound - Wiki. <https://github.com/BloodHoundAD/BloodHound/wiki>, 2018.
- [3] Microsoft. Microsoft Security Compliance Toolkit. <https://docs.microsoft.com/en-us/windows/security/threat-protection/security-compliance-toolkit-10>, 2018.
- [4] Microsoft. Monitoring Active Directory for Signs of Compromise | Microsoft Docs. <https://docs.microsoft.com/en-us/windows-server/identity/ad-ds/plan/security-best-practices/monitoring-active-directory-for-signs-of-compromise>, 2017.
- [5] Microsoft. Appendix L: Events. <https://docs.microsoft.com/en-us/windows-server/identity/ad-ds/plan/appendix-l-events-to-monitor>, 2018.
- [6] JPCERT/CC. LogonTracer. <https://github.com/JPCERTCC/LogonTracer>, 2018.