HSR
HOCHSCHULE FÜR TECHNIK
RAPPERSWIL

COMPUTER SCIENCE

PROJEKTPLAN

Readiness for Tailored Attacks and Lateral Movement Detection

Authors:

Claudio MATTES
claudio.mattes@hsr.ch

Lukas KELLENBERGER
lukas.kellenberger@hsr.ch

Supervisor:

Cyrill BRUNSCHWILER
Compass Security Network Computing AG
cyrill.brunschwiler@compass-security.com

Co-Examiner:

External:
Title Firstname LASTNAME

Internal:
Title Firstname LASTNAME

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

26. September 2018
Version 0.2

Versionshistorie

Version	Datum	Autor(en)	Änderungen
0.1	20.09.18	Claudio Mattes	Initialer Draft
0.2	26.09.18	Team	Erste Formulierung

Inhaltsverzeichnis

1 Einführung

1.1 Zweck

Der Projektplan ist ein Instrument zur Planung, Steuerung und Kontrolle durch das Projektmanagement. Dieses Dokument dient zusätzlich als Beschreibung der Studienarbeit, in welchem das Projekt "Readiness for Tailored Attacks and Lateral Movement Detection" in einer ersten Version umgesetzt wird.

1.2 Gültigkeitsbereich

Der Gültigkeitsbereich beschränkt sich auf die Gesamtdauer des Projektes "Readiness for Tailored Attacks and Lateral Movement Detection" im Modul Studienarbeit des Herbstsemesters 2018. Änderungen werden fortlaufend ergänzt und in der Änderungsgeschichte notiert.

2 Projekt Übersicht

2.1 Ziel und Zweck

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.

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.

2.2 Vorgehen

Das Vorgehen dieser Arbeit ist in einem ersten Schritt eine gründliche Einarbeitung in die Themen

- Incident Handling und Forensik
- Angriffstechniken und Werkzeuge
- Abwehrtechniken und Härtung von Systemen

sowie das Studium öffentlicher Quellen und verfügbarer Tools. Während dieses Prozesses werden noch unbekannte Fragen, wie beispielsweise auf welchen Tools aufgebaut und in welchem Ausmass diese erweitert werden, geklärt. Zudem soll der genau Scope der zu erreichenden Funktionalitäten des Toolkits konkret definiert werden.

2.3 Lieferumfang

- lauffähiges Toolkit und kompletter Source Code
- komplette Software Dokumentation
- komplette Use Cases und Erfolgs-Szenarien resp. Musterlösungen

Die Abgabe des Berichtes erfolgt als gebundenes Dokument. Alle erarbeiteten Daten werden zusätzlich in elektronischer Form auf archiv-i.hsr.ch bereitgestellt.

2.4 Annahmen und Einschränkungen

Es gehört nicht zum Umfang dieser Arbeit neue Angriffsvektoren zu finden. Da dieses Projekt in Form eines Modules mit 8 ECTS Punkten durchgeführt wird, soll dieses mit 2 beteiligten Personen im Rahmen von 480 Arbeitsstunden durchgeführt werden. Das Projekt endet dabei aber spätestens am letzten Studientag es Herbstsemester dem 21. Dezember 2018.

3 Projektorganisation

Das Projekt "Readiness for Tailored Attacks and Lateral Movement Detection" wird im Rahmen der Studienarbeit HS2018 umgesetzt. Dabei besteht das Projekt-Team aus 2 Personen und wird von einem Dozenten betreut.

3.1 Organigramm

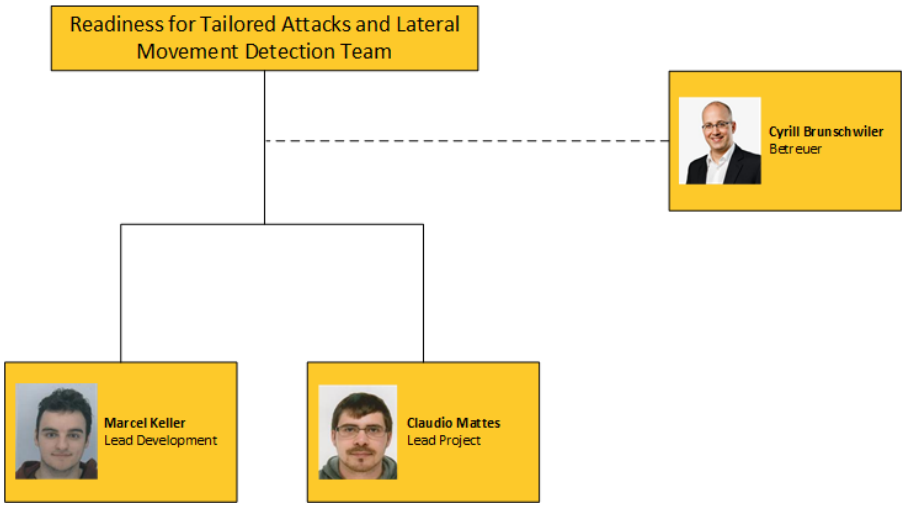

Abbildung 1: Organigramm

3.1.1 Teammitglieder

Nachfolgend sind alle Teammitglieder mit ihren Zuständigkeiten und Kontaktangaben aufgelistet.

Lukas Kellenberger

Kürzel	lkellenb
Email	lkellenb@hsr.ch
Zuständigkeiten	Lead Research and Documentation

Tabelle 2: Angaben Lukas Kellenberger

Claudio Mattes

Kürzel	cmattes
Email	cmattes@hsr.ch
Zuständigkeiten	Lead Project Management and Software Development

Tabelle 3: Angaben Claudio Mattes

3.2 Externe Schnittstellen

Betreuer Ansprechspartner und Betreuer im Rahmen der Studienarbeit HS2018 ist Cyrill Brunschwiler der Compass Security AG.

4 Managementabläufe

4.1 Kostenvoranschlag

Das Projekt wird im Herbstsemester 2018 umgesetzt. Dieses dauert vom 17.09.2018 bis am 21.12.2018. Dabei stehen 14 Semesterwochen zur Verfügung. Pro Woche wird mit einem Zeitaufwand von 34 Stunden[1] pro Woche gerechnet. Dies ergibt rund 480 Stunden über die Gesamtdauer des Projekts. Für Sitzungen sind pro Woche ca. 2 Stunden[2] eingeplant.

4.2 Zeitliche Planung

Grundsätzlich wird während dieser Arbeit nach der Vorgehensweise Scrum+ gearbeitet. Das Projekt wird in vier Phasen Inception, Elaboration, Construction und Transition aufgeteilt. Während den einzelnen Phasen kommt Scrum mit Iterationslängen von 2 Wochen zur Anwendung.

4.2.1 Phasen

Phase	Beschreibung	Dauer
Inception	Kickoff Meeting, Aufgabestellung, Einarbeitung	1 Woche
Elaboration	Projektplan, Meilensteindefinition, Anforderungen, Research	5 Wochen
Construction	Realisierung des Toolkits, Testing, Dokumentation	6 Wochen
Transition	Abschluss Dokumentation, Projektreflexion, Poster	2 Wochen

Tabelle 4: Projektphasen und deren Inhalte

4.2.2 Meilensteinübersicht

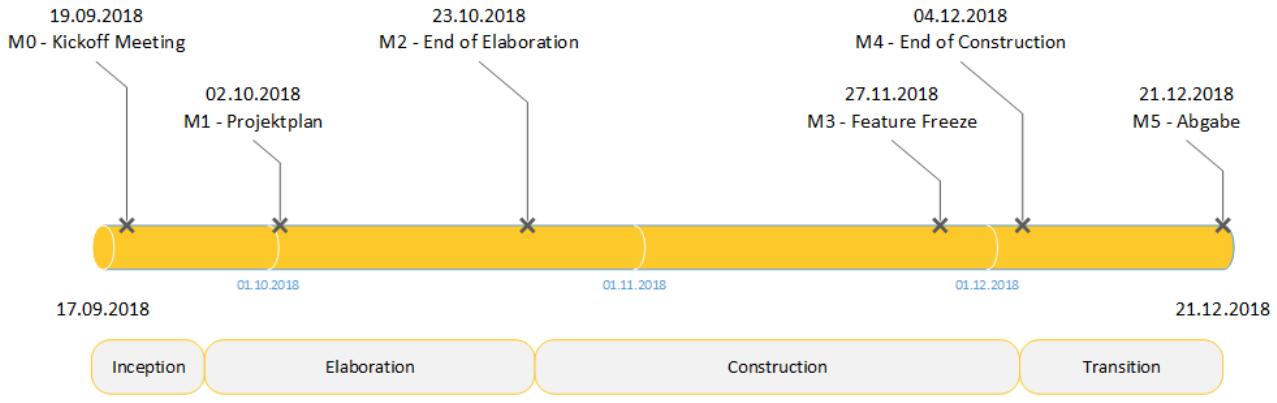

Abbildung 2: Meilensteinübersicht

[1] 17 Stunden pro Person
[2] 1 Stunde pro Woche mit dem Betreuer, sowie 1 Stunde für Iterations- und Teaminterne-Planungen

4.2.3 Meilensteine

Name	Datum	Work Products
M0 – Kickoff Meeting	19.09.2018	• Unterzeichnung Aufgabestellung • Entgegennahme/Besprechung Aufgabestellung
M1 – Projektplan	02.10.2018	• Projektplan in erster Version fertig
M2 – End of Elaboration	23.10.2018	• Abschluss Research • Continuous Integration aufgesetzt • Use Cases definiert • Scope Definition des Toolkits
M3 – Feature Freeze	27.11.2018	• Prototyp mit wichtigste Funktionalitäten erstellt
M4 – End of Construction	4.12.2018	• Alle geplanten Features sind implementiert und getestet & Bugfixes gemacht
M5 – Abgabe	21.12.2018	• Alle Abgabedokumente

Tabelle 5: Meilensteine

4.3 Meetings

Projektmeeting Die Besprechungen mit dem Betreuer finden in der Regel Dienstagvormittag von 08:30 bis 09:30 Uhr statt. Als Ausweichtermin gilt der Donnerstagnachmittag von 13:00 bis 14:00 Uhr.

Teammeeting Das Team trifft sich jeweils Freitagnachmittags zu Besprechungen oder gemeinsamen Reviews.

5 Risikomanagement

6 Arbeitspakete

Die detaillierte Ansicht der Arbeitspakete ist im Redmine ersichtlich. Der Zugriff erfolgt über folgende URL:

- `http://sinv-56085.edu.hsr.ch/redmine/projects/readiness-for-tailored-attacks-and-lateral-movement-detection/issues`

6.1 Aufwand und Schätzung

Der Aufwand wird in Arbeitsstunden angegeben. Diese Stunden sind Schätzungen und können bei auftretenden Problemen oder schnelleren Ausführungen abweichen.

6.2 Priorisierung

Die Arbeitspakete werden wie folgt priorisiert:

Niedrig Feature ist optional, erfolgreicher Projektabschluss hängt nicht von diesem Arbeitspaket ab.

Normal Feature wird im Projekt benötigt, andere Arbeitspakete setzen dieses Feature voraus

Hoch Feature ist zwingend Notwendig für einen erfolgreichen Projektabschluss

Wir fokussieren uns im Projekt auf die Realisierung aller Arbeitspakete, welche die Priorität Hoch sowie Normal erhalten haben. Arbeitspakete welche die Priorität Niedrig erhalten haben, werden bei vorhandener Zeit implementiert. Sollte keine Zeit mehr vorhanden sein, wird dieses Feature mit dem nächsten Release ausgerollt.

7 Qualitätsmassnamen

In diesem Kapitel wird definiert, mit welchen Massnahmen die Qualität des Projekts so hoch wie möglich gehalten wird.

7.1 Dokumentation

Für die Dokumentation besteht ein eigenes Git-Repository. Die Dokumente unterstehen dem Vier-Augen-Prinzip. Dadurch werden Fehler minimiert und die Qualität entsprechend gesteigert.

7.2 Research Dokumente und Tools

Alle verwendeten Dokumente während der Research-Phase werden in einer OneDrive Ablage gehalten. Für die jeweiligen Dokumente und Tools wird jeweils in einer Tabelle eine kurze Zusammenfassung, sowie eine Bewertung der Relevanz für die Studienarbeit aufgeführt.

7.3 Projektmanagement

Als Projektmanagement-Tool haben wir uns für Redmine entschieden, welches sich auf dem virtuellen Server der HSR befindet.

7.3.1 Roadmap

Damit jederzeit die Iteration im Blick gehalten werden kann, bietet Redmine eine Roadmap an. Dort sieht man sofort, wann und welche Iteration erreicht werden muss, sowie auch welche Features diese Iteration beinhaltet und den Status der Features.

Einen weiteren guten Überblick erhält man mit dem in Redmine integrierten Gant-Diagramm.

7.4 Definition of Done

Arbeitspakete werden als abgeschlossen betrachtet, sofern sie folgende Kriterien erfüllen:

- Funktionalität gemäss Beschreibung implementiert
- Feature wurde nachgeführt und geschlossen
- Allfällige Dokumentationen wurden angepasst
- Nachführung von Informationen und Zeiterfassung erledigt
- Issue wurde geschlossen

7.5 Entwicklung

Der Source-Code befindet sich in einem Git-Repository von GitHub, und kann dank Versionierung getrackt werden. Die Qualität wird durch regelmässige Code Reviews und durch das verwenden einer Code Style Guideline sichergestellt.

7.5.1 Unit Testing

Für alle wichtigen Klassen und Komponenten werden systematisch und regelmässig Unit Tests geschrieben. Diese Tests sollen garantieren das der Code fehlerfrei läuft.

7.5.2 Coding Guidelines

- Kein unaufgeräumter (z.B. auskommentierter) Code
- Alles Unfertige ist markiert mit «TODO:»
- Code ist wo nötig & richtig mit Kommentaren versehen

7.5.3 Code Reviews

Die Teammitglieder reflektieren den Code der Anderen in regelmässigen Abständen und geben Verbesserungsvorschläge, welche im Team ausdiskutiert und umgesetzt werden können.

7.5.4 Pair Programming

Je nach Zeitreserven finden auch Pair Programming Sessions statt. Dabei programmiert einer und erläutert laut seine Gedankengänge und der zweite verfolgt die Gedankengänge und Codezeilen, die der andere programmiert. Er meldet sich jeweils, wenn er einen anderen Lösungsweg hat, einen Fehler entdeckt oder auch wenn alles in Ordnung ist.

Abbildungsverzeichnis

Tabellenverzeichnis