# Software Requirements Specification

# For FreeEDR

## Version 5.0.0 approved

#### Prepared by:

|  |  |  |
| --- | --- | --- |
| Bryan Bolesta | Ryan Fiers | Declan Kelly |
| Matthew Horger | Layla Phills | Zachary Santoro |
|  | Marisa Tranchitella |  |

Team FreeEER

April 11, 2020

**Contents**

[1 Introduction 4](#_Toc37507166)

[1.1 Purpose of Document 4](#_Toc37507167)

[1.2 Project Scope 4](#_Toc37507168)

[1.3 Definitions, Acronyms, and Abbreviations 4](#_Toc37507169)

[1.4 Overview 5](#_Toc37507170)

[2 Overall Description 5](#_Toc37507171)

[2.1 Product Functions 5](#_Toc37507172)

[2.2 User Characteristics 5](#_Toc37507173)

[2.2.1 End User 5](#_Toc37507174)

[2.2.2 Script Manager 5](#_Toc37507175)

[2.2.3 Incident Response Manager 5](#_Toc37507176)

[2.2.4 Incident Response Supporter 6](#_Toc37507177)

[2.2.5 System Auditor 6](#_Toc37507178)

[2.2.6 Dashboard Infrastructure Manager 6](#_Toc37507179)

[3 Specific Requirements 6](#_Toc37507180)

[3.1 Functional Requirements 6](#_Toc37507181)

[R1. Server - Rule Access 6](#_Toc37507182)

[R2. Client - Rule Processing 7](#_Toc37507183)

[R3. Dashboard 7](#_Toc37507184)

[R4. Data 8](#_Toc37507185)

[3.2 Non-Functional Requirements 8](#_Toc37507186)

[N1. Hardware 8](#_Toc37507187)

[N3. Deployment 9](#_Toc37507188)

[N4. Other 9](#_Toc37507189)

[3.3 Data Requirements 9](#_Toc37507190)

[D1. Event 9](#_Toc37507191)

[D2. Rule 9](#_Toc37507192)

[D3. Rule Repository 10](#_Toc37507193)

[D4. Network Log 10](#_Toc37507194)

[3.4 Design Constraints 10](#_Toc37507195)

[Bibliography 11](#_Toc37507196)

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Date | Reason For Changes | Version |
| 1.0.0 | 18–10–19 | Initial Structure | mhorger |
| 1.1.0 | 21–10–19 | Design Constraints | dkelly, bbolesta |
| 1.1.1 | 24–10–19 | User Characteristics | lphills |
| 2.0.0 | 27–10–19 | Draft Submission | mhorger |
| 3.0.0 | 03–11–19 | Final Submission | mhorger |
| 4.0.0 | 26-01-20 | Final Submission Cycle 2 | lphills, bbolesta, mhorger, mtranch |
| 5.0.0 | 11-04-2020 | Final Submission Cycle 3 | bbolesta, mhorger, mtranch, zsantor |

# Introduction

## Purpose of Document

The purpose of this software requirement specification document is to outline the functionality and features of our proposed project. The end goal of this document is to assure that the intended audience of FreeEDR understands how FreeEDR will perform, be maintained, and potentially developed further. This document will not describe how the system works in detail or how infrastructure teams should utilize the project. The intended audience of this document is three-fold: first is infrastructure team members of an organization (end-users) interested in implementing our system to enhance their security footprint of workstations. Second is developers who can review the project’s requirements to see where time and research efforts can be utilized to improve the project. Lastly is UAT testers who can test our product when deployed in an environment to find potential security risks or unexpected bugs that stem from requirements.

## Project Scope

FreeEDR is an open-source endpoint detection and response system for small infrastructure teams and organizations to utilize free of charge.

## Definitions, Acronyms, and Abbreviations

* + 1. AD: active directory, a package of special services to manage permissions and resources on Windows workstations
    2. API: application programming interface, technology used for transmitting data between sources such as clients, servers, databases, etc.
    3. EDR: also known as endpoint detection and response, a technology used to address the needs for continuous coverage against advanced threats
    4. Forensic Information: information not included in an event, that assists with analysis. For example, IP address’, process ID’s, file hashes, trust scores, etc.
    5. GPO: group policy object, used when policy settings need to apply to multiple Windows workstations
    6. Sigma: generic, open signature format that allows relevant log events to be reported in a straightforward manner
    7. SigmaC: tool used to translate Sigma format rules to the language of choice
    8. SIEM: Security Information and Event Management
    9. SIRT: Security Incident Response Team
    10. SOC1: also known as a system and organization controls report, used for making sure an organization’s internal control procedures are being properly followed.
    11. Threat Intelligence Sources: security feeds from vendors, government / public, and private sources that provide information about known IT vulnerabilities and risks for organizations.
    12. QA: Quality Assurance

## Overview

Organizations that don’t have a significant security budget can find it difficult to include workstations in their monitoring scope. Forwarding logs from all the organization’s workstations can be expensive because most SIEMs are priced based on log ingestion and tools such as EDR are just as expensive. This project aims to setup a series of scripts which will allow organizations who don’t have the ability to purchase or implement an enterprise solution to monitor workstation traffic on a domain network.

# Overall Description

## Product Functions

* + 1. FreeEDR will generate numerous security alerts against internal network traffic where the system is deployed.
    2. FreeEDR will deploy a secure repository which can update, deliver, and maintain rules that produce said security alerts above.
    3. FreeEDR will allow for internal users to create customizable rules and network processes that are stored on the mentioned secured repository.
    4. FreeEDR will send automatic email and message alerts when security alerts are activated within the system.
    5. FreeEDR will provide an interactive dashboard in order to produce reports, logs, and other auditable information detailing specific time-stamped information.

## User Characteristics

### End User

The End User has their workstation’s network traffic monitored in an organization where FreeEDR is deployed. End Users provide the data and logs that are used by FreeEDR to detect malicious activity and generate alerts. This user will have no privileges to modify or change any aspect of FreeEDR’s deployment in an environment.

### Script Manager

The Script Manager is responsible for the storage, management, and deployment of the provided PowerShell scripts. The script manager must have the privileges and means to deploy and run the scripts on Windows endpoints across the organization. The main function of this user is to deploy the series of scripts to any in-scope endpoints, retain the scripts, and re-deploy scripts when necessary.

### Incident Response Manager

This Incident Response Manager is responsible for ensuring that alerting methods are set-up and properly configured. This user is also responsible for reviewing, responding, and analyzing security alerts. The Incident Response Manager must have view access to the platform that the alerts are being sent (SIEM, Email, or Message). The main function of the Incident Response Manager is to review and respond to the security alerts generated by the FreeEDR scripts.

### Incident Response Supporter

The Incident Response Supporter is responsible for aiding the Incident Response Man- ager in responding to incidents in real time. This user must have the privileges and means to view the reports curated specifically for less technical people. The main function of the incident response supporter is to follow basic instructions to complete tasks that will offset the responsibilities of the Incident Response Manager during an incident response.

### System Auditor

The System Auditor is responsible for auditing the functionality and use of the system. The auditor must have view access of all system and user logs. This user must also have access to any previous audit reports. The main function of the system auditor is to ensure that best practices were followed, the system is being used as effectively as possible and to report any compliance issues discovered to the appropriate parties.

### Dashboard Infrastructure Manager

The Dashboard Infrastructure Manager is responsible for maintaining and deploying future releases of the Audit Dashboard for auditors to view. The infrastructure manager must make sure that the dashboard is consistent with the information that is being produced for the system auditor and have continuous support capacities if some errors were to occur with the reporting tool.

# Specific Requirements

## Functional Requirements

### R1. Server - Rule Access

R1.1. FreeEDR will allow script managers and system administrators full access to sigma rules in order to deploy new changes and update existing content.

R1.2. FreeEDR will allow full access to the sigma rules for Incident Response Managers to view process / network events.

R1.3. FreeEDR will only grant read access to Incident Response Supporters and System Auditors as they should not be able to execute rules.

R1.4. FreeEDR will restrict all access from unauthorized End Users in the organization who have not received internal approval to view the server.

R1.5. FreeEDR server will be allowed to communicate and connect with threat intelligence sources to discover new sigma rules to be stored to the repository.

#### Sigma Rules Permission Matrix

This table represents the Read, Write, and Execute permissions given to users based on their role.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Read** | **Write** | **Execute** |
| *End User* |  |  |  |
| *Script Manager* | x | x | x |
| *Incident Response Manager* | x | x | x |
| *Incident Response Supporter* | x |  |  |
| *System Auditor* | x |  |  |
| *Organization’s System Administrator* | x | x | x |
| *External Sources* |  | x |  |

### R2. Client - Rule Processing

R2.1. FreeEDR will establish a communication channel between clients and the Rule Storage Server to pull down correlation rules.

R2.2. FreeEDR must restrict End Users from writing to the rules repository.

R2.3. FreeEDR will provide a configuration option for the rate at which rules are checked against the event log by the client. This process should be non-disruptive to normal client activity.

R2.4. FreeEDR will establish network security protocols to permit clients to com- municate with APIs.

R2.5. FreeEDR will allow clients to receive information from APIs to perform net- work and process forensics on an event.

R2.6. FreeEDR must be able to store clients’ process and network forensic information within the organization’s filesystem.

R2.7. FreeEDR will store the referenced forensic information for an amount of time set by the organization’s system administrator.

R2.8. FreeEDR must forward forensic events from clients to the Incident Response Manager in order for them to properly perform their user roles.

R2.9. FreeEDR will be able to distinguish which forensics must be applied for dif- ferent client processes and network events.

### R3. Dashboard

R3.1. FreeEDR must allow for communication between the dashboard and the se- cured repository used for rule storage in order to display the list of rules deployed on the dashboard.

R3.2 FreeEDR must grant read access permissions to all users in the business environment so they can request support with any issues. FreeEDR must block all external traffic non-originating from inside the system to the dashboard.

R3.3 FreeEDR, through the dashboard, must allow for the generation of dated reports of when certain events were recorded in the sysmon export.

R3.4 FreeEDR, through the dashboard, must allow for the generation of dated reports of when certain events were recorded for specific users in the sysmon export.

R3.5 FreeEDR, through the dashboard, must allow for the generation of daily reports that details any new rule deployments and new user creations.

R3.6 FreeEDR, through the dashboard, must allow for the generation of a SOC1 report which details any changes to the dashboard or API logic/code.

R3.7 FreeEDR, through the dashboard, must allow for the generation of support tickets and new feature requests from clients.

R3.8. FreeEDR must allow for the Dashboard Infrastructure Manager to allow certain clients to view and/or generate specific reports.

R3.9. FreeEDR will allow clients to select a range of dates for report generation for requirements R3.3 and R3.4.

R3.10. FreeEDR will allow clients to select a specific format to download their gen- erated reports for requirements R3.3 - R3.6.

R3.11. FreeEDR must have an option to export a generated report in requirements R3.3 - R3.6 to send via interdepartmental communication (email, IM, etc).

R3.12. FreeEDR will allow for the dashboard to have a responsive algorithm that allows for regeneration of reports once fresh data is produced. Please see Data Requirements further on the data types.

R3.13. FreeEDR must maintain the dashboard so that it is accessible 90% of normal business hours, with the exception being disaster recovery downtime / failover procedures.

### R4. Data

R4.1. FreeEDR will have an established process in order to track requests, actions, etc in regard to manipulating data in the system.

R4.2. FreeEDR will present Incident Response teams with data on endpoint events (i.e. registry modifications, cross-process events, file executions, network connections).

R4.3. FreeEDR must have 100% uptime access to relevant data sources needed for operations.

R4.4. FreeEDR must keep data for up to 5 years in order to comply with SOC1 reporting / audit procedures. Data past 5 years is outside the jurisdiction of auditable actions may be disposed of.

R4.5. Data transmitted via every API in FreeEDR must be under the proper protocols for security (POST) and sensitive information must be encrypted before transit.

R4.6. FreeEDR should allow the dashboard to access all necessary data in order to produce reports. This data includes forensic event information, user machine configuration, and standard log outputs.

R4.7. FreeEDR must display data in the dashboard in a concise, readable format with an option for details to be viewed separately.

## Non-Functional Requirements

### N1. Hardware

N1.1. FreeEDR will only use hardware in compliance with Security Risk Advisors minimum security requirements.

N1.2. FreeEDR will be deployed on a secure and segmented part of the Drexel CyberDragon’s server until system ownership is transferred.

N2. Network

N2.1. FreeEDR will not use or interact with the Security Risk Advisors network in any way that violates any of their privacy policies.

N2.2. FreeEDR will have an exceptionally high network availability during standard business hours. The target availability for the system is 98.9%.

N2.3. FreeEDR will have an exceptionally high network data retrieval response time during standard business hours. The target response time is within 15 seconds of the client’s request, with larger data requests returning within 5 minutes of the request.

### N3. Deployment

N3.1. FreeEDR will be able to handle all exceptions created by erroneous user input.

### N4. Other

N4.1. FreeEDR’s dashboard must be flexible for modifications in order to add/re- move reports as needed by auditors.

## Data Requirements

### D1. Event

An event is an object captured and displayed by an Event Viewer. These events are triggered by the rules, which are demonstrated below. A sample event is one that has the following data fields:

|  |  |
| --- | --- |
| **Data Field** | **Example Data** |
| Event Type | Critical, Error, Warning |
| Event ID | 111 |
| Source | (Any name of application) |
| Log Location | (Path of application) |
| Date and Time | 11-03-2019 T12:43:00Z |
| Task Category | (Any user defined category) |
| Keywords | (Any user defined keyword) |
| Computer | DESKTOP-341k3 |
| User | SRAPROD/nascoli |
| OpCode | 1 |
| More Information | (Any details go here) |

### D2. Rule

A correlation rule is user-defined logic that can be used to trigger security alerts when endpoints are targeted. For the purposes of FreeEDR, certain data fields can be customized in order to provide modified functionality. An example rule can have the following data fields:

|  |  |
| --- | --- |
| **Data Field** | **Example Data** |
| Title | Suspicious SQL Error Messages |
| Status | experimental |
| Description | Detects SQL error messages that indicate probing for an injection attack |
| Author | Bjoern Kimminich |
| References | <http://www.sqlinjection.net/errors> |
| Logsource | category: application |
| Detetction | keywords |
| keywords | Oracle: quoted string not properly terminated |
| Falsepositives | Application bugs |
| Level | high |

### D3. Rule Repository

A rule repository is a directory structure that can securely store rules in the file format above. Example data structures include:

D3.1. NTFS

D3.2. FAT32

D3.3. Cloud Storage

### D4. Network Log

A network log captures the information when any traffic occurs between clients and endpoints. Network logs typically have the following information:

|  |  |
| --- | --- |
| **Data Field** | **Example Data** |
| Machine Name | DESKTOP-341k3 |
| User Account | SRAPROD/nascoli |
| Date Time of Request | 11-03-2019 T12:43:00Z |
| IP Address | 192.141.53.2 |
| HTTP Response Status Code | 202 OK |
| Headers | Keep-Alive: \* |
| Requested IP Address | 64.10.343.12 |
| Number of Packets Transmitted | 333 |

## Design Constraints

DR1. FreeEDR system architecture must be split in order to establish a server/client relationship for security measures.

DR2. Clients checking for new correlation rules in FreeEDR must be an established fixed interval to allow for appropriate time to gather said rules that have been recently deployed.

DR3. Client process and event information must be stored in a single shared location in order for the FreeEDR’s dashboard to know where to access the information.

DR4. FreeEDR’s reporting dashboard must be hosted on the same server used for rule storage due to security constraints for processing information.

DR5. Correlation rules should be written in Sigma to allow sharing through threat in- telligence platforms such as Threat Alert Logic Repository (TALR)

DR6. FreeEDR correlation rules must be translated to PowerShell Get-Event Queries as this is the supported language in Sigma.

DR7. FreeEDR must be deployed in a Windows environment due to the reliance on Get-Event PowerShell queries.

DR8. FreeEDR must deploy Syson on client workstations to capture the appropriate events for the system to perform properly.

DR9. FreeEDR must satisfy all of Nick Ascoli’s, our external stakeholder for this project, UI preferences when system ownership is transferred to Security Risk Advisors.

# Bibliography

1. Beyond Feeds: A Deep Dive Into Threat Intelligence Sources, *Recorded Future*, h[ttps://www.recordedfuture.com/threat-intelligence-sources/,](http://www.recordedfuture.com/threat-intelligence-sources/) 2019.
2. Threat Alert Logic Repository, *Security Risk Advisors*, https://github.com/SecurityRiskAdvisors/TALR, 2019.