Network Monitor Product Comparison Whitepaper

Parth Bhavsar
Matthew Cotton
Michael Dean
Abhishek Ningala
Jiajia Wu

TABLE OF CONTENTS

EXECUTIVE SUMMARY

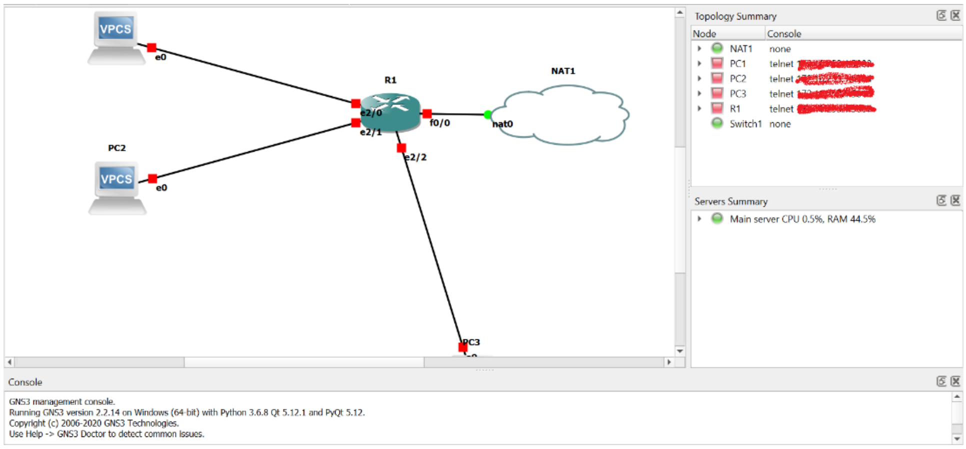
Under the NIST Risk Management Framework (RMF), the Monitor step that follows the authorization step, requires a consistent understanding of actions being taken on and against any device on a certified information system. This requirement was traditionally met on computing systems, but not on network infrastructure or “pure” servers. The goal of this project was to determine the best utility to use in aiding in the monitoring of network devices.

INTRODUCTION TO NETWORK MONITORS AND THEIR USE CASES

This project helps provide some insight into a few aspects of two network auditing tools: Splunk and Datadog. Those aspects include the difficulty in installing and starting the monitors, configuration requirements, and capabilities provided with these tools.

By using GNS3, a network software emulator, our team has created a fully virtualized environment, including a Virtual Router, a lightweight syslog forwarder called Vector, five Virtual Machines on AWS, and individually installed network auditing tools in round to perform testing. Testers will give a rating 1-5 for each section based on their own experience and skill level.

The purpose of this whitepaper is to ensure that an auditing solution meets the requirements of the NIST Risk Management Framework and to provide an effective matrix for selecting the appropriate network monitoring tool for different scenarios.



Node	Console
NAT1	none
PC1	telnet [redacted]
PC2	telnet [redacted]
PC3	telnet [redacted]
R1	telnet [redacted]
Switch1	none

Main server CPU 0.5%, RAM 44.5%

```
GNS3 management console.
Running GNS3 version 2.2.14 on Windows (64-bit) with Python 3.6.8 Qt 5.12.1 and PyQt 5.12.
Copyright (c) 2006-2020 GNS3 Technologies.
Use Help -> GNS3 Doctor to detect common issues.
```

Figure 1- GNS3 Virtual Network

Product 1: Splunk

The first tool we are choosing to assess is Splunk Enterprise. Splunk Enterprise is developed by Splunk Inc., which is an American multinational company located in San Francisco, California. As a product designed to gather data from various sources, Splunk allows security professionals to search, monitor and analyze data via the Splunk Web Interface. Splunk is one of the most popular SIEM tools, hence it was chosen as the first tool for our analysis.

Installation

Installing Splunk is consistently easy. Multiple installation options are available, including via the command-line, via a more user-friendly application package download, and via a cloud-only installation process. The availability of these multiple options generally does not create confusion, and makes the installation of Splunk more flexible.

Splunk's installation process is both well documented by Splunk Enterprise itself, and by the online community of Splunk users, who share their guides and advice on Splunk Enterprise's forums and other popular forums.

The most common issue encountered while installing Splunk, at least onto free-tier AWS EC2 instances, was the minimum free space requirement. While free-tier EC2 instances default to allocating 8 GB of disk space, Splunk requires a minimum of 5 GB of free disk space to ingest logs[todo: cite], and takes up anywhere from 4 to 10 GB for its own software alone. This means that while Splunk will install properly on such a drive, it will not actually function upon startup. No warnings about this are raised during command-line installation, but warnings do appear in the web GUI after Splunk is installed. The simplest remedy is to (re)allocate a larger drive, typically at least 20 GB. This solution is well documented.

Matt: 5/5 – Extremely easy. Well documented. Community documented. Multiple installation options, including: (command-line installation, or download, or cloud setup).
Jiajia: 4/5 – Easy. The path of installation is important, as it can affect the startup. Needs to plan ahead for storage capacity based on the data volume.
Parth: 4/5 – Easy. Lots of documentation available online due to the popularity of the tool. End users might face configuration issues and storage issues, although it isn't too hard to find the solutions.
Abhishek: 5/5 - Extremely easy. User needs to create a Splunk account which is straightforward. Single command to grab and install Splunk.

Startup

Starting Splunk is consistently easy. Starting and stopping Splunk from the command-line consists of running single commands, and the text output of the startup command informs the user of the IP address and port to visit in order to view the Splunk web GUI. The startup and shutdown commands are not immediately documented by Splunk Enterprise from their main installation help page, but are found on other pages, and are community documented. No permissions issues or configuration questions are raised during startup.

In an AWS context, users may forget to allow traffic to port 8000 and/or port 443, which may cause confusion but is easily rectified with AWS security groups. One other expected confusion during this step is locating the startup script after installation, as it may move if Splunk is installed to a non-standard directory or drive, but this issue was uncommon and easily resolved.

Matt: 4/5 – Easy. Not immediately documented. Community documented. Single command. Easy initial configuration.
Jiajia: 4/5 – Easy. Make sure to get the accurate directory to run the command, otherwise the server won't be up.
Parth 5/5 – Extremely Easy. Command was easily found, and the process is pretty streamlined from there.
Abhishek 5/5 - Extremely Easy. Best practice to run as Splunk user. Simple commands to start and stop Splunk

Configuration / Data Connection

Configuring Splunk can be moderately confusing. Several steps are required to set up TCP or UDP log ingestion, and these steps span multiple pages and submenus. Users are required to provide some specific details about the type of data being ingested (syslogs, files, HTTP traffic, etc.), but confusingly these configuration options only seem to affect the Splunk Indexer, and not the actual ingestion. This may confuse or dissuade users from setting up otherwise simple log forwarding. This log ingest configuration process is not well documented by Splunk Enterprise, but is covered by user-written guides. Even the more network-competent testers had to consult user-written guides in this regard.

One that may arise, especially in an AWS context, is in forgetting to allow inbound TCP and/or UDP traffic to Splunk server. This is easily rectified with AWS security groups, and this solution is well documented.

Matt: 3/5 – Moderate. Several steps and settings are required to enable TCP log ingest.
Jiajia: 3/5 – Moderate. Specific details including port number, source type and static IP address are required to ingest data through TCP.
Parth: 3/5 – Moderate. Steps can be followed by using a documented guide but might require some level of expertise and have caused ingestion errors.
Abhishek: 3/5 - Moderate. Need to ensure files are owned by Splunk user and has data access to certain files (e.g., /var/log). Need to add the rule to open port 8000 in the AWS security group for Splunk Web UI. Storage issues may arise.

Validation and Usage

It can range from easy to difficult to use Splunk and validate that it is functioning, depending on the user’s familiarity with SIEM tools. Novice testers had difficulty searching and querying for logs, and several testers had trouble in determining whether logs were being received at all, as the default Splunk search page does not show any logs by default (a search term must be entered and run, first). For more experienced users, however, the learning curve is shallow, and the tools available on-screen allow for powerful and specific queries to be written easily in the SPL (“search processing language”) syntax.

Some useful features of Splunk include its automatic parsing and indexing of JSON-formatted logs, which improves search and readability. By default, the entire contents of logs appear onscreen, which can be overwhelming by itself without searching and filtering.

Matt: 3/5 – Moderate. Hard to tell when logs actually come in -- have to query for something. Page shows all detail up-front, can be overwhelming. Automatic JSON parsing.
Jiajia: 2/5 – Difficult. A certain level of SQL knowledge is required to use Search Processing Language. Search commands will be complex when dealing with a huge amount of data.
Parth: 4/5 – Easy. Has a bit of learning curve, but from personal experience, it is my tool of choice. Extremely powerful once you can correctly identify filters like index and sourcetype.
Abhishek: 3/5 - Need to apply time range and proper filters to identify the logs coming in. Basic understanding of SPL is required to get insightful data.

Self-Reflection Matrix

Tester	Installation	Startup	Configuration	Validation	Self-Reflection (competence, tool suitability)
Matt	5/5	4/5	3/5	3/5	4/5, 3/5
Jiajia	4/5	4/5	3/5	2/5	3/5, 3/5
Parth	4/5	5/5	3/5	4/5	4/5, 4/5
Abhishek	5/5	5/5	3/5	3/5	5/5, 3/5

Product 2: Datadog

We are choosing Datadog as the second product for our testing. Datadog is a SaaS-based data analytics platform headquartered in New York City, United States. It provides services like Security Monitoring, Network Monitoring and Incident Management.

Installation

Installing Datadog is extremely easy. Being a SaaS (cloud-only) service, and requiring only a Datadog account, there is no server configuration or access necessary for installation. The process is well documented by Datadog. The only initial confusion for a novice user may be in the requirement of a Datadog account before installation, but this is easily clarified by documentation.

Matt: 4/5 – Easy. Well documented. Requires cloud account. Cloud-only.
Jiajia: 5/5 – Extremely easy. Set up cloud account. SaaS-based platform.
Parth: 5/5 – Extremely easy. Setup just requires an account and the API key. Don't even need to install on the AWS EC2 Instance.
Abhishek: 5/5 - Extremely easy. Single command to install Datadog. Purely SaaS alternative

Startup

Starting up the Datadog service is trivial, as no actions are required. Datadog continuously manages and runs the service for its users on its own servers, so the service is always on. No space or memory issues are encountered for new and low-usage installations, and no initial configuration is necessary whatsoever.

Matt: 5/5 – Extremely easy. Occurs automatically, always-on. Datadog runs the server for you. No space or memory issues. No initial configuration.
Jiajia: 5/5 – Extremely easy. No command/action is required.
Parth: 5/5 – Extremely easy. No startup required since it is cloud based. Always on.
Abhishek: 5/5 - Extremely easy. Always up and running

Configuration / Data Connection

Configuration of Datadog is generally very easy. In order to set up TCP log forwarding, users must simply acquire their API key, which is found (very logically) via the Integrations tab of the Datadog web GUI's main page. Beyond that, there is just a single, static, public Datadog URL and port to which users may send any and all logs for ingest. Datadog handles the parsing and indexing of all ingested logs without user configuration necessary.

The only initial holdup for novice users may be in understanding this process, but the log ingest process is itself documented on the API key page, and is furthermore explained by Datadog's own documentation, and by the community. In an AWS context, the only technical issue that may arise on the log forwarder's side is in allowing traffic outbound on the requisite ports. This is easily resolved using AWS security groups.

Matt: 4/5 – Easy. Only API key is required, URL and ports are public and static. API keys are simple to locate and use. Well documented. Have to know what search terms to use to find this feature.
Jiajia: 4/5 – Easy. Logs and metrics can be received through HTTP by adding API keys in log forwarder. No need to configure anything else.
Parth: 5/5 – Extremely easy. Just requires adding the API key in the log forwarder and it works.
Abhishek: 5/5 - Extremely Easy. Easy to locate and use API key. Numerous integrations available.

Validation and Usage

Validating that Datadog is functioning and using it are generally easy. The log page is easily found, and both auto-refreshes automatically and performs an initial search without requiring user input. This makes it very easily to look for test logs, and to see new logs arrive in real-time.

By default, only the Date, Host, Service, and Content of logs are displayed on-screen without being clicked, which reduces visual clutter. However, some testers felt that too many other controls and details were shown by default, cluttering the UI. JSON-formatted logs are also automatically parsed, which improves search and readability.

Matt: 4/5 – Easy. Easy to tell when logs come in. Log page is not overloaded with detail. Can drill into logs for more detail. Automatic JSON parsing.
Jiajia: 4/5 – Easy. The logs page can show up new logs without querying them. Not automatically display stats count as percentage.
Parth: 4/5 – Extremely easy. Interface is easy to use, though might feel cluttered. Moderate amount of documentation available.
Abhishek: 4/5 - Easy. Insightful event stream to look at most recent updates. Easy to visualize at more granular level.

Self-Reflection Matrix

Tester	Installation	Startup	Configuration	Validation	Self-Reflection (competence, tool suitability)
Matt	4/5	5/5	4/5	4/5	4/5, 4/5
Jiajia	5/5	5/5	4/5	4/5	4/5, 4/5
Parth	5/5	5/5	5/5	4/5	4/5, 5/5
Abhishek	5/5	5/5	5/5	4/5	3/5, 4/5

CONCLUSION

After the testing of our two tools, it was made clear that while Splunk is the preferred tool for a more technical person, if it’s possible to use a SaaS solution, Datadog has a more streamlined setup and integration. If an information system can support a SaaS solution (not a closed environment) the easier solution would be Datadog.

However, if your environment would like more defined, advanced searches, the dashboard building function of Splunk provides the ability to build specific search functions and display the data in a very easy to decipher format, if you have somebody with the skills to establish the dashboards.

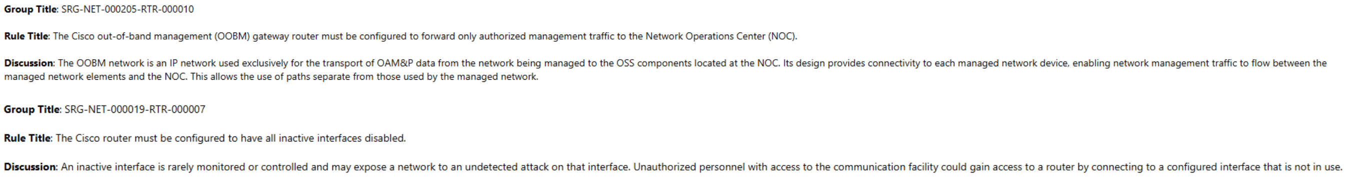
TESTING METHODOLOGY

The testing was performed in a standard approach for each tool. The following events were generated:

1. Failed login for operator account (3x)
2. Successful login for administrative account
3. New device added to network
4. Old device taken off network
5. Interface shutdown
6. Interface turned on

These events are the general events that would be monitored in a certified information system and meet most security control requirements laid forth by Defense Information Systems Agency (DISA) to meet RMF compliance.

Examples:



Group Title: SRG-NET-000205-RTR-000010

Rule Title: The Cisco out-of-band management (OOBM) gateway router must be configured to forward only authorized management traffic to the Network Operations Center (NOC).

Discussion: The OOBM network is an IP network used exclusively for the transport of OAM&P data from the network being managed to the OSS components located at the NOC. Its design provides connectivity to each managed network device, enabling network management traffic to flow between the managed network elements and the NOC. This allows the use of paths separate from those used by the managed network.

Group Title: SRG-NET-000019-RTR-000007

Rule Title: The Cisco router must be configured to have all inactive interfaces disabled.

Discussion: An inactive interface is rarely monitored or controlled and may expose a network to an undetected attack on that interface. Unauthorized personnel with access to the communication facility could gain access to a router by connecting to a configured interface that is not in use.

Further testing down the road should include port-security settings that would allow for VLAN hopping, sticky-mac configuration, and MAC protection. This testing was unable to be performed during this phase, due to an older iOS that would require an additional license purchase to upgrade.