**NECCDC 2021 – Log Management Documentation**

**Elastic Stack & Wazuh**

**Alexander Zimmer**

**Pace University - BergCyberSec**

**[INITIAL TASKS]**

Things that can be done when disconnected from the internet. This will be done in conjunction with initial manual threat hunting.

**Threat Hunting**

*Delete pre-installed installed SEIMs and associated directories.*

Splunk: /opt/splunk and /opt/SplunkForwarder

Wazuh: /var/ossec

Elastic Stack: /usr/share/elasticsearch, /usr/share/kibana, /usr/share/filebeat,

Graylog: /usr/share/graylog-server

*Check crontab configurations for unexpected script triggering.*

/etc/crontab

/var/spool/cron/crontabs

*Check for any malicious aliases of commands in /etc and individual user home directories.*

/etc/bashrc or /etc/bash.bashrc

/home/<user>/.bashrc

**iptables Configuration**

Implement iptables-service and Disable firewalld. Need sudo privileges.

systemctl disable firewalld

yum -y (apt-get) install iptables-services

systemctl enable iptables

systemctl start iptables

service iptables save

iptables -F

Ensure iptables Rules Persist

nano /etc/sysconfig/iptables-config

IPTABLES\_SAVE\_ON\_STOP="yes"

IPTABLES\_SAVE\_ON\_RESTART="yes"

SSH Input

sudo iptables -A INPUT -p tcp --dport 22 -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT

SSH Output

sudo iptables -A OUTPUT -p tcp --sport 22 -m conntrack --ctstate ESTABLISHED -j ACCEPT

Subnet Input Rule

iptables -A INPUT -s <192.168.1.0/24> -p <tcp/udp> --dport <1514/1515/5601> -m state --state

NEW,ESTABLISHED -j ACCEPT

Subnet Output Rule

iptables -A OUTPUT -d <192.168.1.0/24> -p <tcp/udp> --sport <1514/1515/5601> -j ACCEPT

Save iptables Rules

sudo iptables-save

sudo service iptables save

RELEVANT PORTS TO ALLOW

1514 (tcp & udp): Wazuh Agent Events

1515 (tcp): Wazuh Agent Registration

514 (tcp & udp): Syslog Events (potentially not needed)

5601 (tcp): Kibana Web Interface (May be best to specify the IP address that can access rather than subnet)

55000 (tcp): Wazuh HTTP Requests (May be best to not have this accessible outside the host.)

View iptables Rules

sudo iptables -S

Flush All iptables Rules

sudo iptables -F

**BASH Command Logging**

The filename will be different depending on whether it is a RHEL or Debian based Linux distribution.

sudo nano /etc/bashrc *or* sudo nano /etc/bash.bashrc

Enter the following code as a single line entry at the bottom of the file.

PROMPT\_COMMAND='LAST\_COMMAND=$(history 1 | sed 's/-/\\-/') && logger -i -p local5.info -t bash "$USER $(tty): $LAST\_COMMAND"'

Reload .bashrc parameters to start command logging.

sudo . /etc/bashrc (bash.bashrc)

**[POST CONNECTION]**

Tasks that can be done once the network firewall reconnects to the internet.

**Installation**

All-in-One Installation with the Elastic Stack Basic License. Add -i at the end to force install if there are limited system resources.

curl -so ~/all-in-one-installation.sh https://raw.githubusercontent.com/wazuh/wazuh-documentation/4.0/resources/elastic-stack/unattended-installation/all-in-one-installation.sh && bash ~/all-in-one-installation.sh -i

The script will run a health check, which can be stopped with -i or --ignore-healthcheck

Save the password generated for the ‘elastic’ user to later access Kibana.

**Configuration**

Defaults to 0.0.0.0, which projects the interface to the full subnet. Best to specify the destination ip address.

*/etc/kibana/kibana.yml*

server.host: <ip\_address>

Make sure filebeat has the proper credentials. Flip archives to true to allow for all log collection.

*/etc/filebeat/filebeat.yml*

username: “elastic”

password: “<elastic\_password>”

* module: wazuh

alerts:

enabled: true

archives:

enabled: true

Flip logall and logall\_json to yes to allow analysis of all log content. Syscheck frequency sets the interval for file integrity checks based on seconds.

*/var/ossec/etc/ossec.conf*

<global>

<logall>yes</logall>

<logall\_json>yes</logall\_json>

</global>

<syscheck>

<frequency>60</frequency>

</syscheck>

Store the password in a variable so it isn’t recorded in command history as plain text.

read -s PASS

Generate Wazuh API SSL Certificate using the current user password.

TOKEN=$(curl -u <user>:<password> -k -X GET "https://localhost:55000/security/user/authenticate?raw=true")

Store the needed JSON data structure containing the new password ($PASS) in a new variable.

JSON=’{ “password”: “’“$PASS”’”, “allow\_run\_as”: false }’

Change User Password: Based off user ID #

curl -k -X PUT "https://localhost:55000/security/users/1" -H "Authorization: Bearer $TOKEN" -H "Content-Type: application/json" -d “$JSON”

Update password in wazuh configuration file.

/usr/share/kibana/optimize/wazuh/config/wazuh.yml

Change Elastic Search Password. This method is for the basic license, not for the opensource version.

curl -k -X POST -u <username>:<user\_password> "https://<elasticsearch\_ip>:9200/\_security/user/<username>/\_password?pretty" -H 'Content-Type: application/json' -d '{ "password" : "<new\_password>"}'

Returns empty json structure {} if successful. Can alternatively be changed in the GUI, but elasticsearch service needs to be restarted.

Update the password in associated files.

/etc/filebeat/filebeat.yml and /etc/kibana/kibana.yml

**Start and Check systemctl Services**

Each requires sudo permissions.

systemctl enable <service>

systemctl start/restart <service>

systemctl status <service>

Services: [iptables-services, elasticsearch, kibana, filebeat, wazuh-manager]

**Access Kibana**

Depends on whether the interface is being projected and if the firewall allows the port traffic. Could also be localhost.

URL: https://<wazuh\_server\_ip>

user: elastic

password: <PASSWORD\_elastic>

Set up the search parameter for the archive index.

Stack Management > Index Patterns > Create Index Pattern

wazuh-archives-\*

Set as default for custom searches and refresh the fields list.

**[Dashboards]**

**Command Stream**

BASH Command Stream

(location:/var/log/messages OR location:/var/log/syslog) AND “bash”

Fields: [full\_log, agent.name, agent.id, predecoder.timestamp]

This depends on each participating Linux system having added the proper line of code the /etc/bashrc or /etc/bash.bashrc file and reloading the file into memory or re-logging into the system. (from earlier instructions)

Powershell Command Stream

data.win.system.eventID:4104

Fields: [data.win.eventdata.scriptBlockText, agent.id, agent.name, data.win.system.process, @timestamp]

Dependent on the Windows Domain Controller or all Windows devices adjusting their group policy. Powershell Module Logging and Script Block Logging both need to be enabled. Powershell must also be listed in the modules.

Computer Configuration > Administrative Templates > Windows Components > Windows Powershell

Module Logging: Enable

Options: Show…

Microsoft.Powershell.\*

PowerShell Script Block Logging: Enable

* Log Script Block Invocation Start / Stop Events

All windows computers need to update their group policies in Powershell

Gpupdate

**Searches**

Successful User Logon

"session opened" OR "accepted password" OR "accepted publickey" OR data.win.system.eventID:4624

Failed User Logon

"authentication failure" OR "failed password" OR data.win.system.eventID: 4625

User Logoff

"session closed" OR data.win.system.eventID: 4634

Sudo / Privlaged Actions

sudo OR "FAILED su"

Log Tampering

data.win.system.eventID:4719 OR data.win.system.eventID:1102 OR "bashrc"

Service Installation

data.win.system.eventID:4697 OR "apt-get" OR "apt install" OR "wget" OR "yum install"

User Modification

"usermod" OR "useradd" OR "adduser" OR data.win.system.eventID:4720 OR data.win.system.eventID:4722 OR data.win.system.eventID:4725 OR data.win.system.eventID:4738 OR data.win.system.eventID:4740 OR data.win.system.eventID:4767

Group Modification

"groupadd" OR "groupmod" OR "addgroup" OR data.win.system.eventID:4735 OR data.win.system.eventID:4737 OR data.win.system.eventID:4755

Zeek/Bro Logs

location: /opt/bro/spool/manager/\*.log