ISSN: 2614-6754 (print)
ISSN: 2614-3097(online)

Information Security Systems Design Using SIEM, SOAR and Honeypot

Muhammad Hafiz[1], Benfano Soewito[2]
[1,2] Computer Science Department Binus Graduate Program – Master of Computer Science Universitas Bina Nusantara
e-mail: muhammad.hafiz003@binus.ac.id[1], bsoewito@binus.edu[2]

Abstrak

Informasi merupakan aset yang sangat berharga bagi suatu organisasi atau perusahaan karena merupakan salah satu sumber daya strategis dalam meningkatkan nilai bisnis. Oleh karena itu, perlindungan informasi (information security) merupakan hal mutlak yang harus diperhatikan secara serius oleh seluruh jajaran dalam suatu organisasi, mulai dari tingkat manajemen puncak dan seluruh karyawan di setiap levelnya. XYZ memiliki ancaman serangan siber dari peretas yang berusaha mengakses dan memiliki aset penting dan rahasia. Untuk melindungi aset tersebut, diperlukan sistem keamanan yang dapat melindungi dari berbagai cara atau teknik serangan dari pihak yang tidak bertanggung jawab. Dibutuhkan sistem keamanan informasi berlapis untuk dapat mendeteksi dan merespon serangan siber yang terjadi secara otomatis. Security Information and Event Management (SIEM) adalah alat untuk sentralisasi log, korelasi, pelaporan, dan peringatan. Security Orchestration, Automation and Response (SOAR) adalah teknologi yang dapat merespons insiden serangan siber secara otomatis. Selain itu, dibutuhkan juga sistem yang menyerupai sistem aslinya untuk menjaga jika seorang penyerang atau hacker dapat melewati security perimeter atau Firewall, sistem ini disebut Honeypot. Hasil pengujian setelah penerapan SIEM, SOAR dan Honeypot pada PT XYZ merupakan sistem yang dapat memantau serangan siber yang muncul secara real-time, merespon secara otomatis sehingga keamanan informasi dapat terlindungi secara optimal.

Kata kunci: *Security Information and Event Management; Security Orchestration Automation and Response; Honeypot; Keamanan Cyber; Keamanan Informasi*

Abstract

Information is an unbelievably valuable asset for an organization or company because it is one of the strategic resources in increasing business value. Therefore, the protection of information (information security) is an absolute must be considered seriously by all ranks in an organization, starting from the top management level and all employees at every level. PT. XYZ has a cyberattack threat from hackers who seek to access and possess important and confidential assets. To protect these assets, a security system is needed that can protect against many ways or techniques of attack from irresponsible parties. It takes a layered information security system to be able to detect and respond to cyberattacks that occur automatically. Security Information and Event Management (SIEM) is a tool for centralization of logs, correlate, reporting and alerting. Security Orchestration, Automation and Response (SOAR) is a technology that can respond to cyberattack incidents automatically. Furthermore, it also takes a system that resembles the original system to guard if an attacker or hacker can pass through the security perimeter or Firewall, this system is called Honeypot. Test results after the application of SIEM, SOAR and Honeypot on PT. XYZ is a system that can monitor cyberattacks that appear in real-time, responding automatically so that information security can be protected optimally.

Keywords: *Security Information and Event Management; Security Orchestration Automation and Response; Honeypot; Cyber Security; Information Security*

INTRODUCTION

Information security has three aspects, namely process, people, and technology. Information security is an asset for organizations that when implemented properly will provide convenience to improve competence while increasing the possibility for the successfulness of work activities and business processes conducted. To ensure that the organization's business processes are in accordance with the principles of useful information security, it is necessary to implement information security processes using methodology based on standard best practices that are commonly used and applicable internationally.

The National Cyber and Crypto Agency (BSSN) has published Honeynet Project Annual Report in 2018, on that report there were 12,895,554 cyber attacks that entered Indonesia [1]. Meanwhile, next year in 2019, there were 98,243,896 total cyber attacks detected into Indonesia [2].

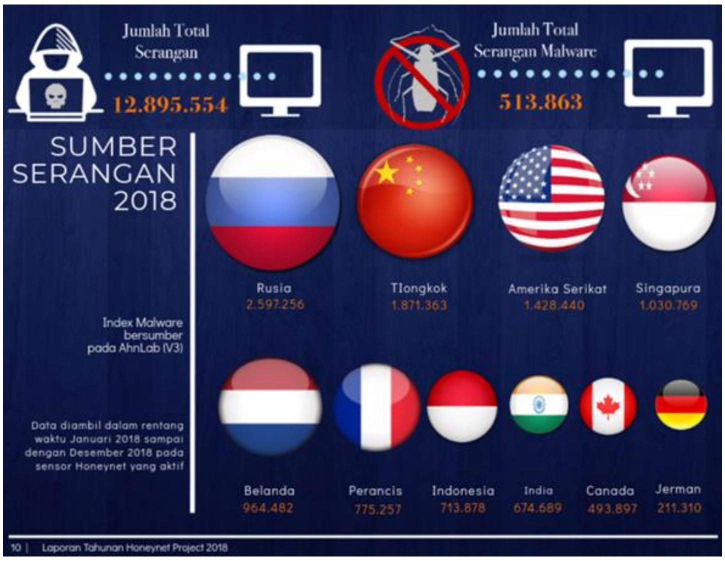


Figure 1 Recapitulation of cyber attacks 2018
Source : [1]

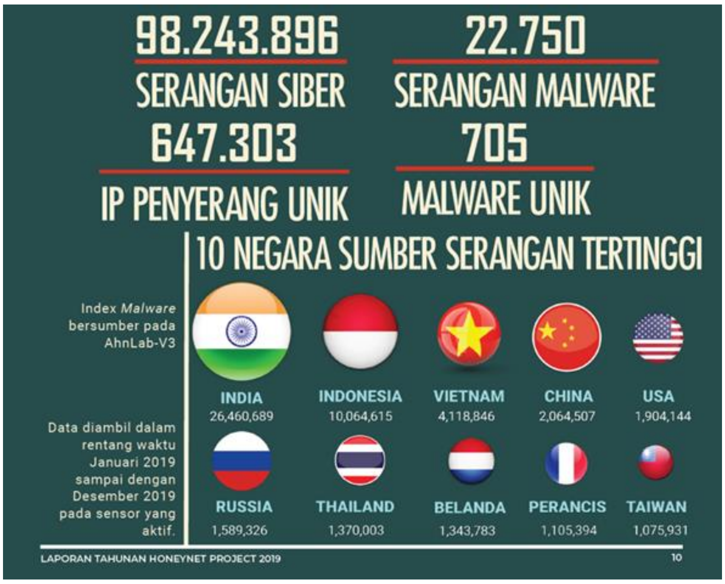


Figure 2 Recapitulation of Cyber Attacks 2019
Source: [2]

METHODE

In this paper, the method used by the authors in figure 3 is to combine three security systems that have complementary functions, namely Security Information and Event Management (SIEM) is used to detect anomaly activity and also as a place to centralize logs. Furthermore, Security Orchestration, Automation and Response (SOAR) is a technology that can automate and organize tasks, processes, policy execution, and reporting. Furthermore, a system that resembles the original system to guard if an attacker or hacker can bypass the security perimeter or Firewall, this system is called Honeypot is implemented at a general insurance company. This paper obtaining data sources from literature studies. The literature study used is obtained from previous research, scientific journals, books, and e-books. In conducting research, it must be able to describe the interrelationship of each stage or process so that activities in research become more focused, directed, and systematic therefore the author in the design activities of this security system adopts the Network Development Life Cycle (NDLC) in figure 4. According to [3] Network Development Life Cycle (NDLC) is a method used to develop or design computer network systems and allows monitoring of systems that are being designed or developed in order to be known for performance.

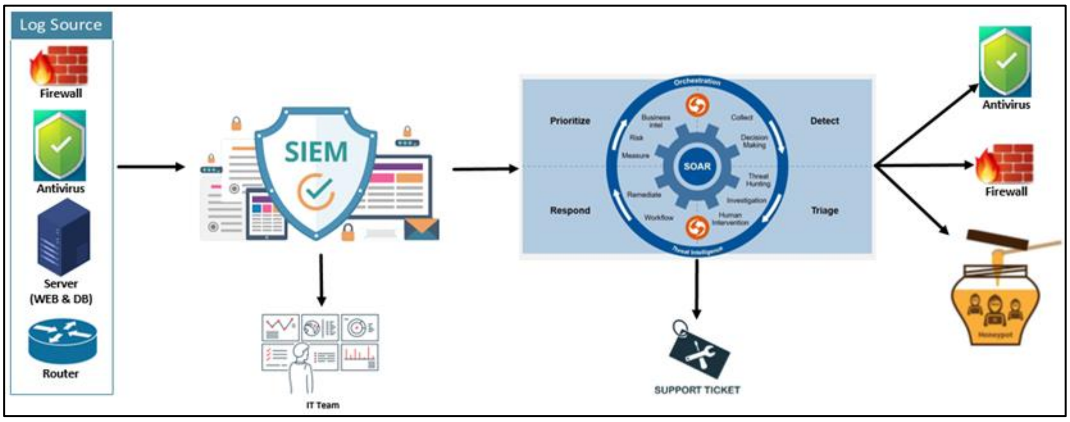


Figure 3 Proposed Information Security System

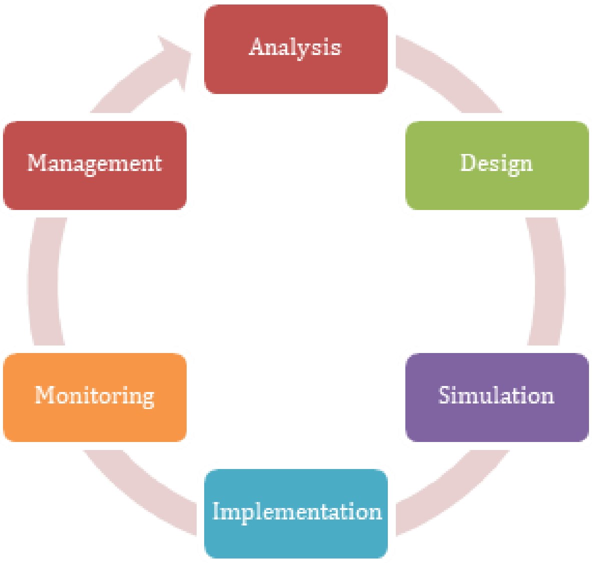


Figure 4 Network Development Life Cycle Method

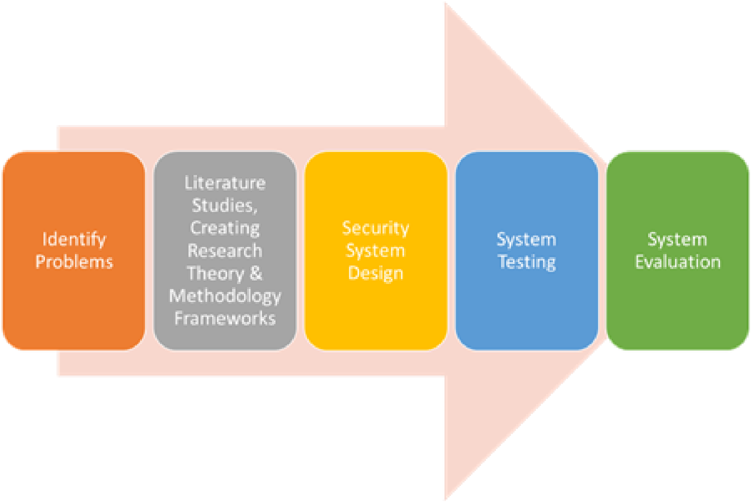


Figure 5 Research Framework

Stages of Research

In making this thesis, the author will add a layer to the existing security system PT XYZ. The research steps to be conducted are divided into five steps, namely problem identification, literature study, making a theoretical framework & research methodology, Security System Design, System Testing and System Evaluation in figure 5.

1. Identify Problems
   At this stage, identification of the problems that often arise in the company is conducted. This is done by holding meetings and interviews with the IT division of the company, with the hope that the author can obtain data related to problems faced by the company and the data needed in designing the security system.
2. Literature Studies, Creating Research Theory & Methodology Frameworks
   In the next step the author conducts a literature study or exploration of the theory relevant to the problems faced by other companies. Literature studies are conducted to understand the theory as well as related information to solve problems derived from the research that has been conducted.
3. Security System Design
   The next stage is the design that will be made or proposed by the author based on the problems that exist in the company and input from the IT division so that the design of the security system made is right on target, as well as documents containing data obtained from the previous stage.
4. System Testing
   At this stage, testing is conducted at the same time with monitoring, to find out whether the design is in accordance with what is expected by the company. In addition, testing was conducted, how effective the SIEM, SOAR and Honeypot methods are for the automation of securing systems and data from cyberattacks.
5. System Evaluation
   At this stage, the author makes an evaluation plan for the design of the security system that has been built. By using several tools such as Low Orbit Ion Cannon (LOIC) to perform DoS which can interfere with services (services) on existing operating systems in the company because it targets resources (CPU and Memory) and utilizes tools found in the Kali Linux operating system, such as SQL Map (SQL Injection), Nikto or Nmap (Scanning port). Simulations were conducted twice, namely before and after implementing a security system using SIEM, SOAR and Honeypot. The following table serves as a reference to evaluate the system to be conducted by the author.

Table 1 Security System Testing Evaluation Results

Attack Simulation Conditions	Type of Attack			Information
	DoS	Port Scanning	SQL Injection	
Before the implementation of SIEM, SOAR and Honeypot				
After the implementation of SIEM, SOAR and Honeypot				

RESULTS AND DISCUSION

It is explained that the topology of the attack flow conducted by hackers goes directly to the destination server. After implementing the proposed security system, attacks that go directly to the server system will be reduced because the attack will be detected by SIEM that gets logs or events from the firewall which then provides notifications to the administrator containing the hacker's IP address, destination IP and the type of attack used. SIEM forwards to SOAR to process to the next stage such as blocking from antivirus or forward network traffic to Honeypot server

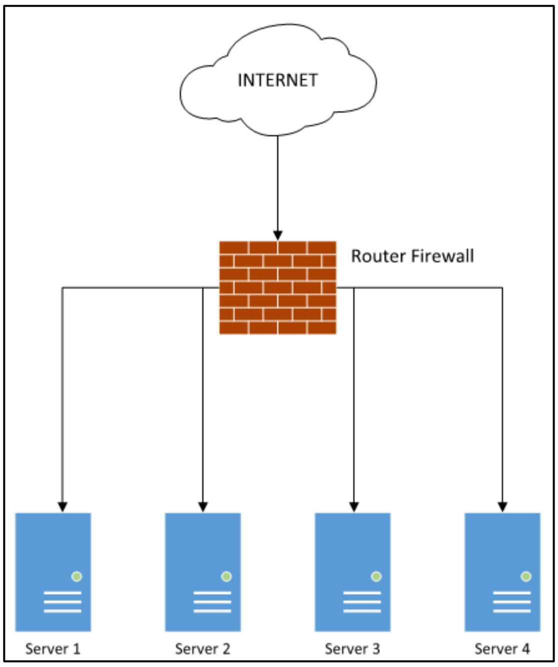


Figure 6 Topology before implementation security system

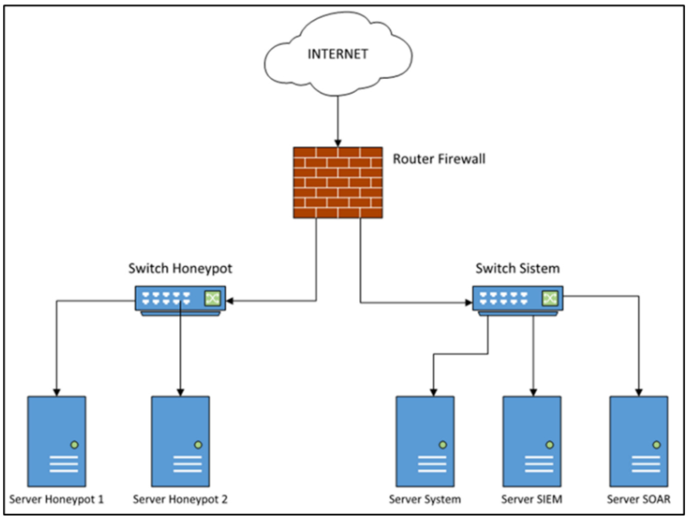
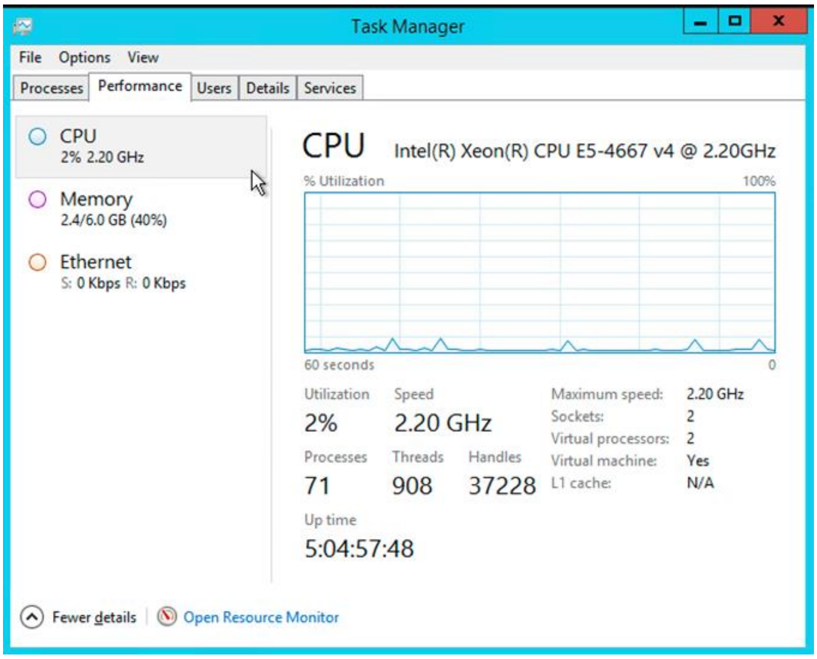


Figure 7 Topology after implementation security system

RESULTS BEFORE SECURITY SYSTEM IMPLEMENTATION

Before assessing the simulation of cyberattacks to the system, the authors checked the server to be able to compare the results when simulating the attack.



CPU
2% 2.20 GHz

Memory
2.4/6.0 GB (40%)

Ethernet
S: 0 Kbps R: 0 Kbps

CPU Intel(R) Xeon(R) CPU E5-4667 v4 @ 2.20GHz

% Utilization 100%

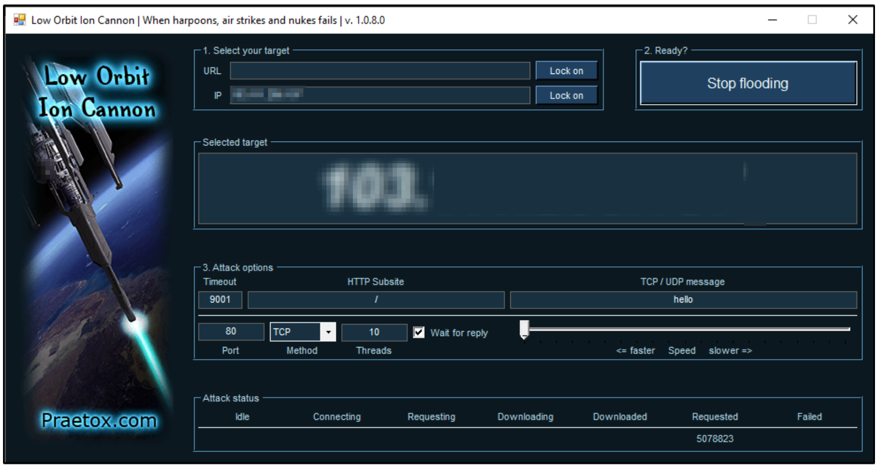
60 seconds 0

Utilization 2%
Speed 2.20 GHz
Processes 71
Threads 908
Handles 37228
Up time 5:04:57:48

Maximum speed: 2.20 GHz
Sockets: 2
Virtual processors: 2
Virtual machine: Yes
L1 cache: N/A

Figure 8 Server condition before simulated cyberattacks

Figure 8 represents the normal state of the server when it is operational. It can be seen from the CPU, Memory and Ethernet showing conditions that are running normally. Furthermore, the author conducted a test with a simulation of a DoS attack directly on the targeted server to see the conditions when the attack occurred.



Low Orbit Ion Cannon | When harpoons, air strikes and nukes fails | v. 1.0.8.0

1. Select your target
URL
IP [redacted]

2. Ready? Stop flooding

Selected target [redacted]

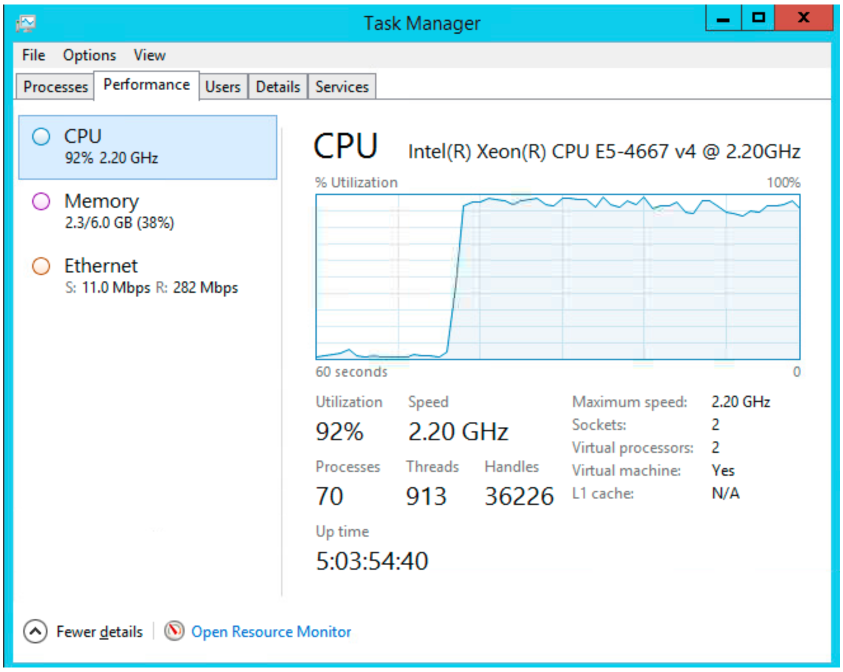
3. Attack options
Timeout: 9001
HTTP Subsite: /
TCP / UDP message: hello
Port: 80
Method: TCP
Threads: 10
☑ Wait for reply
<= faster Speed slower =>

Attack status

Idle	Connecting	Requesting	Downloading	Downloaded	Requested	Failed
					5078823	

Figure 9 Simulated DoS attacks directly to the target server

In figure 9, the simulation of a DoS attack was conducted by the author using the Low Orbit Ion Cannon (LOIC) tool by targeting the company's server. Where this is extremely dangerous because it can interfere with the availability of services in the system used so that it has an impact on operations and losses that can be experienced by the company.



Task Manager

CPU 92% 2.20 GHz
Memory 2.3/6.0 GB (38%)
Ethernet S: 11.0 Mbps R: 282 Mbps

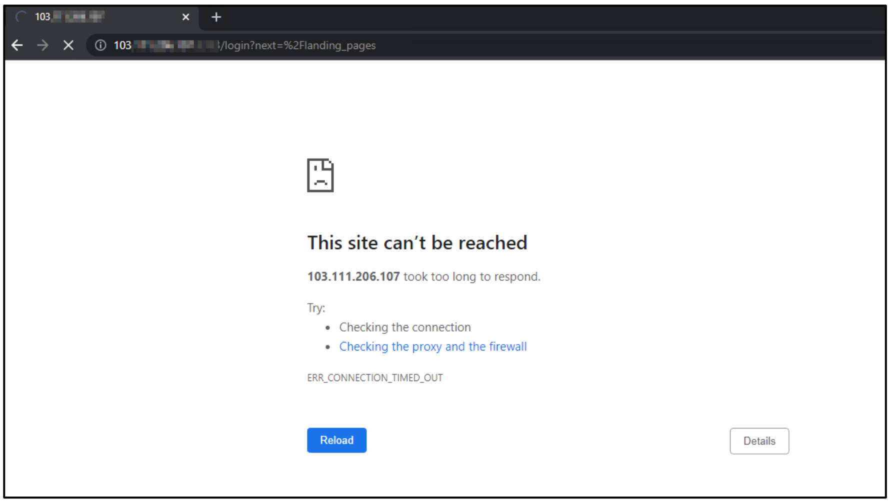
CPU Intel(R) Xeon(R) CPU E5-4667 v4 @ 2.20GHz

Utilization 92%
Speed 2.20 GHz
Processes 70
Threads 913
Handles 36226
Up time 5:03:54:40

Maximum speed: 2.20 GHz
Sockets: 2
Virtual processors: 2
Virtual machine: Yes
L1 cache: N/A

Figure 10 Server status when DoS attack simulation running

In figure 10, you can see an increase in CPU usage during a DoS attack. This happens because there is no security system in place to detect, respond and trick hackers. So that the server becomes an easy target that can be used by hackers to interfere with services on the company's systems.



This site can't be reached

103.111.206.107 took too long to respond.

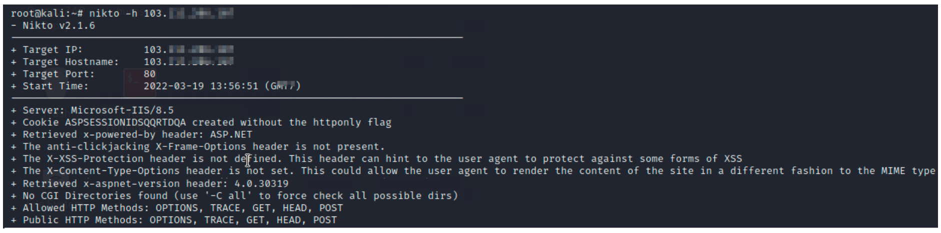
Try:

- Checking the connection
- Checking the proxy and the firewall

ERR_CONNECTION_TIMED_OUT

Figure 11 System condition when a cyberattack occurs

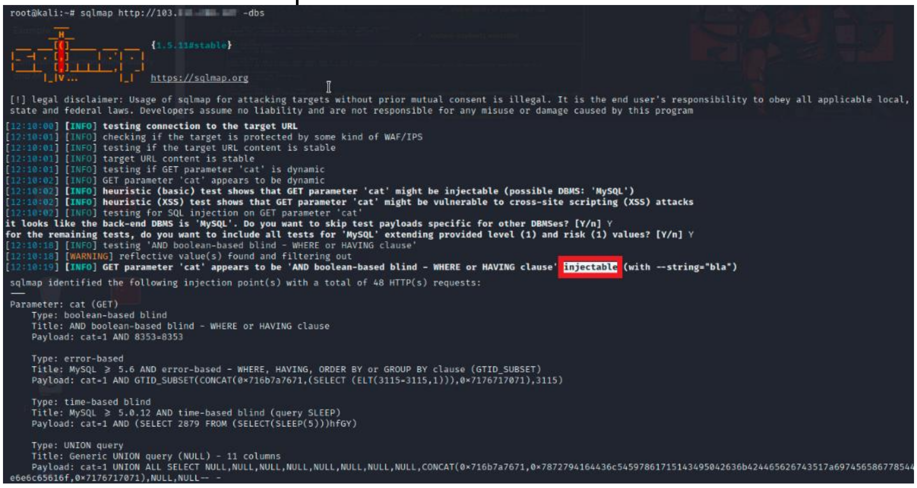
When a DoS attack occurs, the company's system cannot work normally because it is affected by the DoS attack. This can be detrimental to the company because it could be that the system being attacked is the main system of the company.



```
root@kali:~# nikto -h 103.[redacted]
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          103.[redacted]
+ Target Hostname:    103.[redacted]
+ Target Port:        80
+ Start Time:         2022-03-19 13:56:51 (GM[redacted])
---------------------------------------------------------------------------
+ Server: Microsoft-IIS/8.5
+ Cookie ASPSESSIONIDSQQRTDQA created without the httponly flag
+ Retrieved x-powered-by header: ASP.NET
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type
+ Retrieved x-aspnet-version header: 4.0.30319
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ Allowed HTTP Methods: OPTIONS, TRACE, GET, HEAD, POST
+ Public HTTP Methods: OPTIONS, TRACE, GET, HEAD, POST
```

Figure 12 Attack simulation using Port Scanning

In the figure 12 can be seen, the author simulated port scanning using the Nikto tool where the tool can be directly used on the Kali Linux operating system. This can inform hackers because the result of this attack is in the form of a port or service active on the system, which can lead the hacker to conduct a follow-up attack to obtain other information.



```
root@kali:~# sqlmap http://103.[redacted] -dbs

                              {1.5.11#stable}

                              https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local,
state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program

[12:10:00] [INFO] testing connection to the target URL
[12:10:01] [INFO] checking if the target is protected by some kind of WAF/IPS
[12:10:01] [INFO] testing if the target URL content is stable
[12:10:01] [INFO] target URL content is stable
[12:10:01] [INFO] testing if GET parameter 'cat' is dynamic
[12:10:02] [INFO] GET parameter 'cat' appears to be dynamic
[12:10:02] [INFO] heuristic (basic) test shows that GET parameter 'cat' might be injectable (possible DBMS: 'MySQL')
[12:10:02] [INFO] heuristic (XSS) test shows that GET parameter 'cat' might be vulnerable to cross-site scripting (XSS) attacks
[12:10:02] [INFO] testing for SQL injection on GET parameter 'cat'
it looks like the back-end DBMS is 'MySQL'. Do you want to skip test payloads specific for other DBMSes? [Y/n] Y
for the remaining tests, do you want to include all tests for 'MySQL' extending provided level (1) and risk (1) values? [Y/n] Y
[12:10:18] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[12:10:18] [WARNING] reflective value(s) found and filtering out
[12:10:19] [INFO] GET parameter 'cat' appears to be 'AND boolean-based blind - WHERE or HAVING clause' injectable (with --string="bla")
sqlmap identified the following injection point(s) with a total of 48 HTTP(s) requests:
---
Parameter: cat (GET)
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
    Payload: cat=1 AND 8353=8353

    Type: error-based
    Title: MySQL >= 5.6 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (GTID_SUBSET)
    Payload: cat=1 AND GTID_SUBSET(CONCAT(0x716b7a7671,(SELECT (ELT(3115=3115,1))),0x7176717071),3115)

    Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
    Payload: cat=1 AND (SELECT 2879 FROM (SELECT(SLEEP(5)))hfGY)

    Type: UNION query
    Title: Generic UNION query (NULL) - 11 columns
    Payload: cat=1 UNION ALL SELECT NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,CONCAT(0x716b7a7671,0x7872794164436c54597861715143495042636b4244656267435l7a697456586778544
e6e6c65616f,0x7176717071),NULL,NULL-- -
```

Figure 13 SQL Injection Attack Simulation

In this test, the author launched a SQL Injection attack using the SQL Map tool where the *tool* can be directly used on the Kali Linux operating system. The effect of this attack is that hackers can enter the system without having to use a username and password, and then can modify the data stored in the Database which results in invalidated existing data.

Results After Implementation Of Siem, Soar And Honeypot Security Systems

This stage describes when an attack goes to a server that previously hackers could go directly to the target server, after the implementation of the attack can be detected by the SIEM by informing the source of the hacker's IP, the destination IP, and the activity or type of attack used in the process.

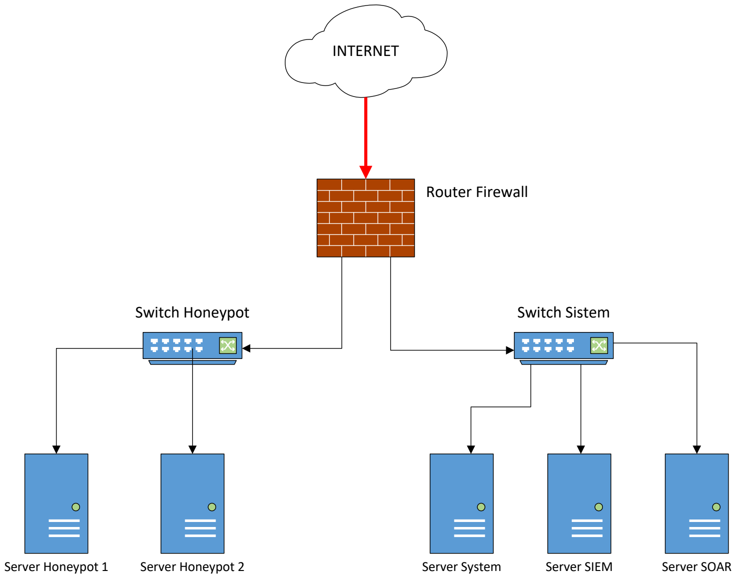


Figure 14 Topology when blocking attacks

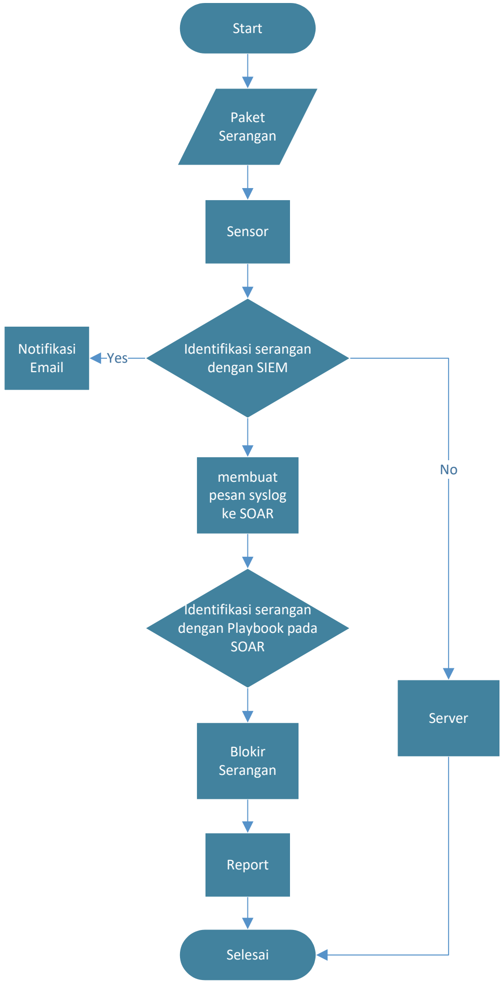
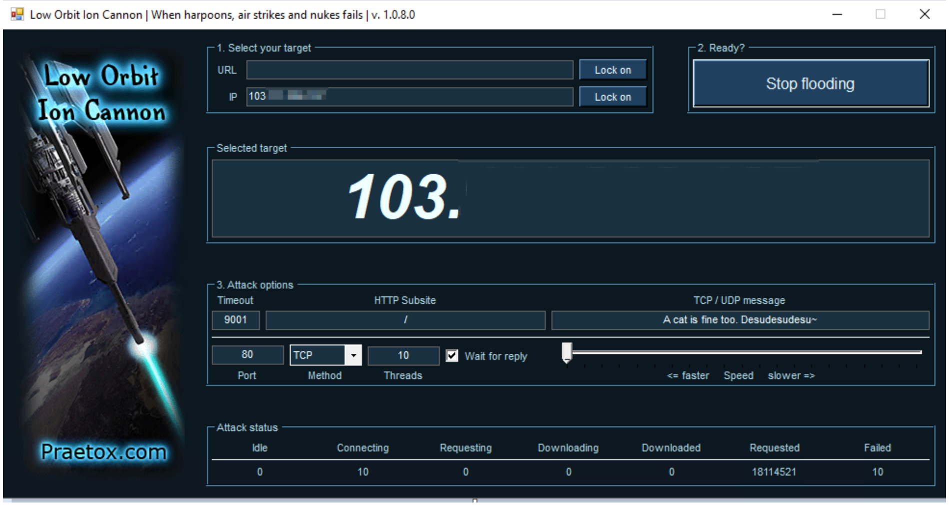

Start
Paket Serangan
Sensor
Identifikasi serangan dengan SIEM
Yes
Notifikasi Email
No
membuat pesan syslog ke SOAR
Identifikasi serangan dengan Playbook pada SOAR
Server
Blokir Serangan
Report
Selesai

Figure 15 Flow when blocking attacks

Figure 14 is the topology when an attack block occurs and figure 15 is an attack blocking flow conducted by a firewall device where in this process the SIEM receives logs and correlates them from the sensors contained in the firewall. Monitoring is conducted on every incoming (inbound) and outbound package. If an incoming attack is detected, the SIEM will provide an email notification and forward the logs to SOAR in order to order the firewall to block attempted attacks that enter the company.


Low Orbit Ion Cannon | When harpoons, air strikes and nukes fails | v. 1.0.8.0
Low Orbit Ion Cannon
Praetox.com
Lock on
Lock on
Stop flooding
<= faster   Speed   slower =>

1. Select your target
URL
IP 103 [redacted]

2. Ready?

Selected target
103.

3. Attack options

Timeout	HTTP Subsite	TCP / UDP message
9001	/	A cat is fine too. Desudesudesu~

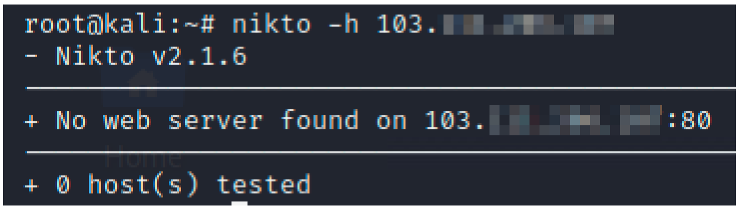
Port	Method	Threads	
80	TCP	10	☑ Wait for reply

Attack status

Idle	Connecting	Requesting	Downloading	Downloaded	Requested	Failed
0	10	0	0	0	18114521	10

Figure 16 DoS simulation when autoblock is applied

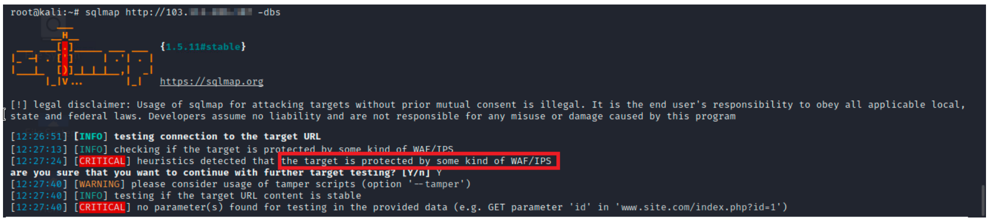
Figure 16 proves that when the security system has been implemented, the attack process can be detected and blocked automatically.



```
root@kali:~# nikto -h 103.[redacted]
- Nikto v2.1.6
---------------------------------------------------------------
+ No web server found on 103.[redacted]:80
---------------------------------------------------------------
+ 0 host(s) tested
```

Fiigure 17 *port scanning simulation when autoblock is applied*

Figure 17 proves that the simulation to detect the active port on the target system was unsuccessful because it had been detected by the security system implemented so that hackers could not have information from the targeted system to be hacked.



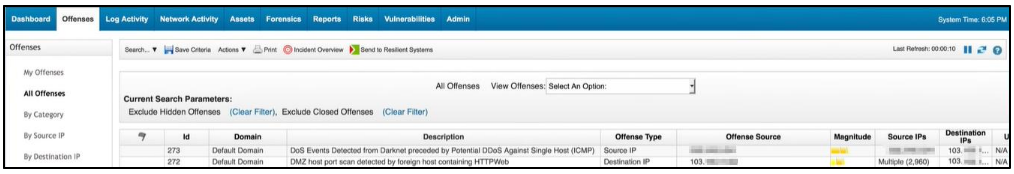
```
root@kali:~# sqlmap http://103.[redacted] -dbs
        ___
       __H__
 ___ ___[.]_____ ___ ___  {1.5.11#stable}
|_ -| . [']     | .'| . |
|___|_  [)]_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local,
state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program

[12:26:51] [INFO] testing connection to the target URL
[12:27:13] [INFO] checking if the target is protected by some kind of WAF/IPS
[12:27:24] [CRITICAL] heuristics detected that the target is protected by some kind of WAF/IPS
are you sure that you want to continue with further target testing? [Y/n] Y
[12:27:40] [WARNING] please consider usage of tamper scripts (option '--tamper')
[12:27:40] [INFO] testing if the target URL content is stable
[12:27:40] [CRITICAL] no parameter(s) found for testing in the provided data (e.g. GET parameter 'id' in 'www.site.com/index.php?id=1')
```

Figure 18 *SQL Injection simulation when autoblock is applied*

From the attempted simulation of the attack using SQL *Injection* in figure 18, it can be found that the system has been protected by WAF / IPS. This happens because the security implementation has detected and responded to incoming attacks. So that attacks from hackers do not reach the intended server.

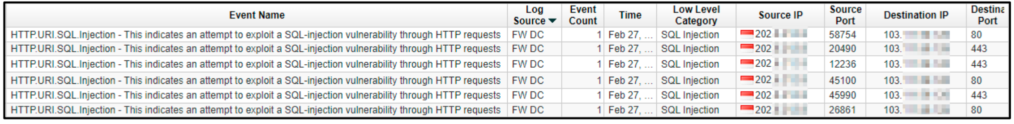


All Offenses View Offenses: Select An Option:

Current Search Parameters:
Exclude Hidden Offenses (Clear Filter), Exclude Closed Offenses (Clear Filter)

	Id	Domain	Description	Offense Type	Offense Source	Magnitude	Source IPs	Destination IPs	U
	273	Default Domain	DoS Events Detected from Darknet preceded by Potential DDoS Against Single Host (ICMP)	Source IP	[redacted]		[redacted]	103.[redacted] ...	N/A
	272	Default Domain	DMZ host port scan detected by foreign host containing HTTPWeb	Destination IP	103.[redacted]		Multiple (2,960)	103.[redacted] ...	N/A

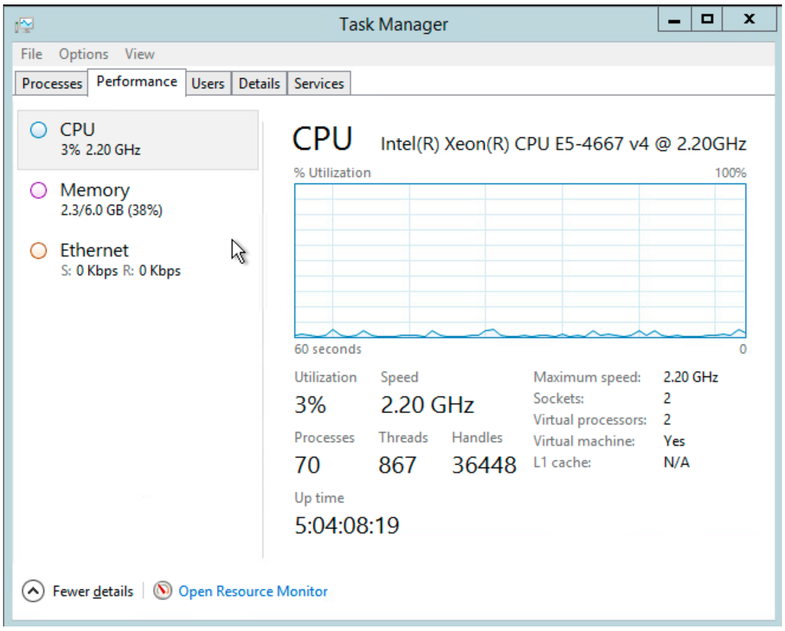
Figure 19 Log of Simulated DoS and Port Scan results detected by SIEM



Event Name	Log Source ▼	Event Count	Time	Low Level Category	Source IP	Source Port	Destination IP	Destina Port
HTTP.URI.SQL.Injection - This indicates an attempt to exploit a SQL-injection vulnerability through HTTP requests	FW DC	1	Feb 27, ...	SQL Injection	202[redacted]	58754	103.[redacted]	80
HTTP.URI.SQL.Injection - This indicates an attempt to exploit a SQL-injection vulnerability through HTTP requests	FW DC	1	Feb 27, ...	SQL Injection	202[redacted]	20490	103.[redacted]	443
HTTP.URI.SQL.Injection - This indicates an attempt to exploit a SQL-injection vulnerability through HTTP requests	FW DC	1	Feb 27, ...	SQL Injection	202[redacted]	12236	103.[redacted]	443
HTTP.URI.SQL.Injection - This indicates an attempt to exploit a SQL-injection vulnerability through HTTP requests	FW DC	1	Feb 27, ...	SQL Injection	202[redacted]	45100	103.[redacted]	80
HTTP.URI.SQL.Injection - This indicates an attempt to exploit a SQL-injection vulnerability through HTTP requests	FW DC	1	Feb 27, ...	SQL Injection	202[redacted]	45990	103.[redacted]	443
HTTP.URI.SQL.Injection - This indicates an attempt to exploit a SQL-injection vulnerability through HTTP requests	FW DC	1	Feb 27, ...	SQL Injection	202[redacted]	26861	103.[redacted]	80

Figure 20 Log SQL Injection detected by SIEM

In figures 19 and 20 the attacks conducted by hackers were detected by siem, the information obtained was the type of attack, the hacker's IP, the destination IP, the destination port. Then the information is forwarded to SOAR to order the firewall to block the attack. So that the server or system targeted by the hacker does not experience interference as before.



CPU
3% 2.20 GHz

Memory
2.3/6.0 GB (38%)

Ethernet
S: 0 Kbps R: 0 Kbps

CPU Intel(R) Xeon(R) CPU E5-4667 v4 @ 2.20GHz

Utilization 3%
Speed 2.20 GHz
Processes 70
Threads 867
Handles 36448
Up time 5:04:08:19

Maximum speed: 2.20 GHz
Sockets: 2
Virtual processors: 2
Virtual machine: Yes
L1 cache: N/A

Figure 21 Server conditions when an attack is blocked by a firewall

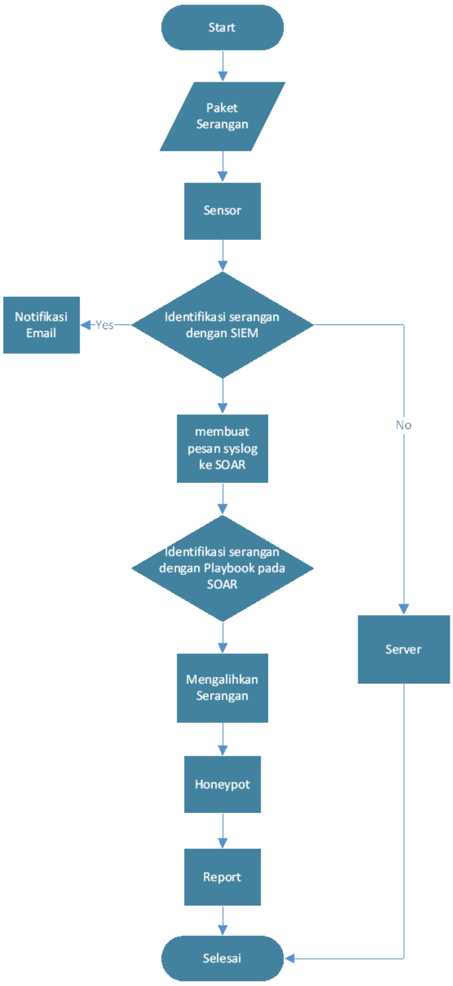

Start
Paket Serangan
Sensor
Identifikasi serangan dengan SIEM
Yes
Notifikasi Email
No
membuat pesan syslog ke SOAR
Identifikasi serangan dengan Playbook pada SOAR
Server
Mengalihkan Serangan
Honeypot
Report
Selesai

Figure 22 Flow when switching attacks

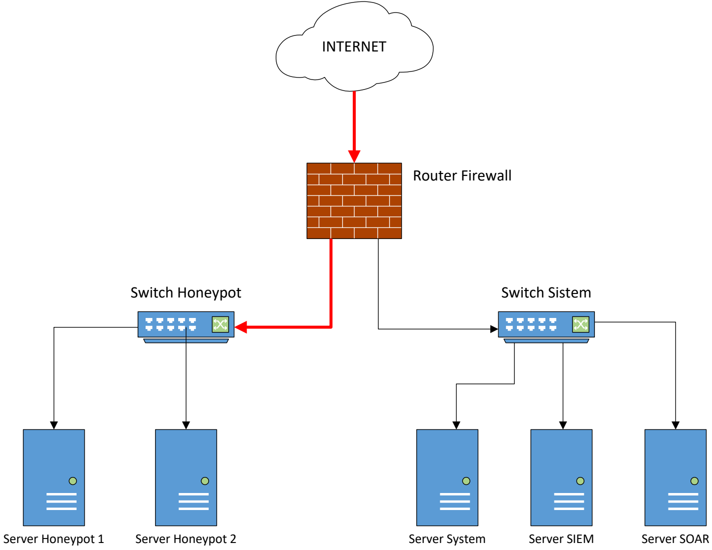
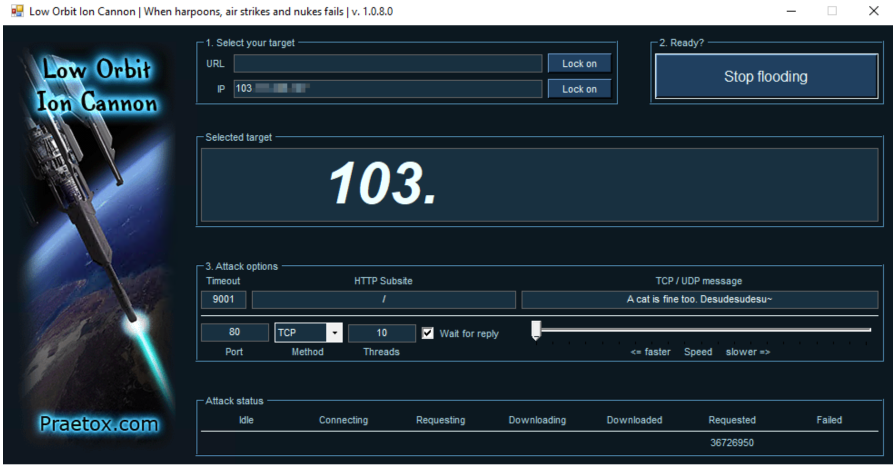


Figure 23 Topologies when redirecting attacks

Figure 22 is the blocking flow of the attack and figure 23 is the topology when the attack redirect occurs that the SIEM performs in this process is to receive *logs* and correlate them from the received sensors. Monitoring is conducted on every incoming (*inbound*) and outbound package. Then if an incoming attack is detected, the SIEM will provide an email notification and forward the *logs* to SOAR so that it can order the *firewall* to divert to a *honeypot* server that has been prepared as a fake server to learn and trick hackers from attempted attacks that enter the company.


Low Orbit Ion Cannon | When harpoons, air strikes and nukes fails | v. 1.0.8.0
Low Orbit Ion Cannon
Praetox.com
Lock on
Lock on
Stop flooding

1. Select your target
URL
IP 103 [redacted]

2. Ready?

Selected target
103.

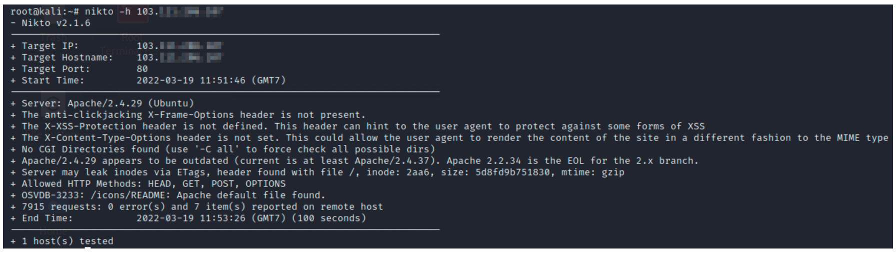
3. Attack options
Timeout: 9001
HTTP Subsite: /
TCP / UDP message: A cat is fine too. Desudesudesu~
Port: 80
Method: TCP
Threads: 10
☑ Wait for reply
<= faster Speed slower =>

Attack status

Idle	Connecting	Requesting	Downloading	Downloaded	Requested	Failed
					36726950	

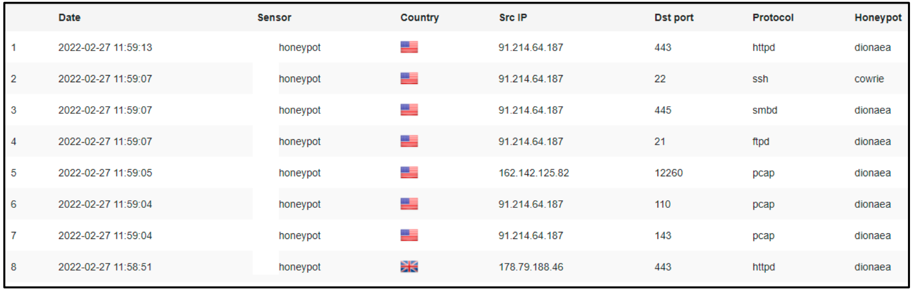
Figure 24 DoS simulation when automatic redirection is applied

The DoS simulation performed in figure 24 is still running because a security system has been implemented to divert the chain from hackers, this will not make hackers feel suspicious because it is considered that the attack remains connected to the destination system.



```
root@kali:~# nikto -h 103.[redacted]
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          103.[redacted]
+ Target Hostname:    103.[redacted]
+ Target Port:        80
+ Start Time:         2022-03-19 11:51:46 (GMT7)
---------------------------------------------------------------------------
+ Server: Apache/2.4.29 (Ubuntu)
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ Apache/2.4.29 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2.34 is the EOL for the 2.x branch.
+ Server may leak inodes via ETags, header found with file /, inode: 2aa6, size: 5d8fd9b751830, mtime: gzip
+ Allowed HTTP Methods: HEAD, GET, POST, OPTIONS
+ OSVDB-3233: /icons/README: Apache default file found.
+ 7915 requests: 0 error(s) and 7 item(s) reported on remote host
+ End Time:           2022-03-19 11:53:26 (GMT7) (100 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

When the attacker conducts an attack attempt on a company that has implemented SIEM, SOAR and Honeypot security systems. The attack can be detected so that the attack conducted by hackers can be redirected to a fake server (Honeypot) can be seen with the type of server currently using ubuntu where the original server or system uses the *Windows* operating system (figure 8) so that the server is not affected by the attack. The redirects made will not make hackers immediately aware of this because the attack process is still ongoing.



	Date	Sensor	Country	Src IP	Dst port	Protocol	Honeypot
1	2022-02-27 11:59:13	honeypot	🇺🇸	91.214.64.187	443	httpd	dionaea
2	2022-02-27 11:59:07	honeypot	🇺🇸	91.214.64.187	22	ssh	cowrie
3	2022-02-27 11:59:07	honeypot	🇺🇸	91.214.64.187	445	smbd	dionaea
4	2022-02-27 11:59:07	honeypot	🇺🇸	91.214.64.187	21	ftpd	dionaea
5	2022-02-27 11:59:05	honeypot	🇺🇸	162.142.125.82	12260	pcap	dionaea
6	2022-02-27 11:59:04	honeypot	🇺🇸	91.214.64.187	110	pcap	dionaea
7	2022-02-27 11:59:04	honeypot	🇺🇸	91.214.64.187	143	pcap	dionaea
8	2022-02-27 11:58:51	honeypot	🇬🇧	178.79.188.46	443	httpd	dionaea

Figure 4. 1 Logs on the Honeypot

The log of the attack conducted is caught by the honeypot, the information that can be obtained, namely, the IP used, the country that conducted the attack and the port or service that is trying to be utilized.

CONCLUSION

Based on the analysis of the Implementation of Cyber Security Systems using SIEM, SOAR and Honeypot on PT. XYZ can conclude as follows:

1. Cyber security systems are designed using dual security systems, between (SIEM) and Honeypot, to effectively secure critical assets from companies. Successfully designed by utilizing devices owned by the company to be able to protect the system and data on PT. XYZ.
2. Based on the results of testing cyber security systems using SIEM, SOAR and Honeypot have reliability and capabilities in maintaining cyber security. Proven security systems can cope with attacks on testing by using Sql Injection, DOS, and Port Scanning.

Table 2 Security System Testing Evaluation Results

Attack Simulation Conditions	Type of Attack			Information
	DoS	Port Scanning	SQL Injection	
Before the implementation of SIEM, SOAR and Honeypot	Succeed	Succeed	Succeed	Attacks from attackers can directly attack the destination server.
After the implementation of SIEM, SOAR and Honeypot	Fail	Fail	Fail	Attacks from Attackers can be detected by SIEM and can be blocked and redirected to Honeypot.

Based on the test results of the implementation of the security system using *Security Information Event Management* (SIEM), *Security Orchestration, Automation and Response* (SOAR) and Honeypot at PT XYZ. It can be concluded that the system can monitor in real time and can detect, respond automatically and redirect attacks to fake servers prepared to trick hackers. From the simulation of the three techniques such as DoS, Port Scanning and SQL Injection can effectively secure assets such as data and systems in the company.

REFERENCES

Amarullah, I., & Kurniawan, M. T. (2017, Agustus). Perancangan Jaringan Multi-Protocol Label Switching Menggunakan Metode NDLC Untuk Layanan File Transfer Protocol Dan Web Service Universitas Telkom. e-Proceeding of Engineering, 4(2), 3099-3106. Retrieved 2021, from https://openlibrarypublications.telkomuniversity.ac.id/index.php/engineering/article/view/1302/1244

Granadillo, G. G., El-Barbori, M., & Debar, H. (2016, November 21-23). New types of Alert Correlation for Security Information and Event Management Systems. 2016 8th IFIP International Conference on New Technologies, Mobility and Security (NTMS), 1-7. doi:10.1109/NTMS.2016.

IBM. (n.d.). topics:siem. Retrieved Desember 2020, from https://www.ibm.com/topics/siem

Kinyua, J., & Awuah, L. (2021). AI/ML in Security Orchestration, Automation and Response: Future Research. Intelligent Automation & Soft Computing, 28, 527-545. doi:10.32604/iasc.2021.016240

Kinyua, J., & Awuah, L. (2021). AI/ML in Security Orchestration, Automation and Response: Future Research. Intelligent Automation & Soft Computing, 28, 527-545. doi:10.32604/iasc.2021.016240.

Mohammad, S. M., & Lakshmisri, S. (2018). Security automation in Information technology. International Journal of Creative Research Thoughs (IJCRT), 6(2), 901-905. Retrieved from https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3652597.

mythixy. (2017, Maret). Retrieved from https://netsec.id/honeypot/

Neiva, C., Lawson, C., Bussa, T., & Sadowski, G. (2020, September 21). documents: 3990720. Retrieved from Gartner: https://www.gartner.com/en/documents/3990720

Nilă, C., Apostol, I., & Patriciu, V. (2020). Machine learning approach to quick incident response. 2020 13th International Conference on Communications (COMM), 291-296. doi:10.1109/COMM48946.2020.9141989.

Prasetya, N. I., Djanali, S., & Husni, M. (2014). Verifikasi Signature Pada Kolaborasi Sistem Deteksi Intrusi Jaringan Tersebar Dengan Honeypot. Jurnal Ilmiah Teknologi Informasi, 12(2), 70-82. Retrieved from https://sci-ub.ru/https://www.academia.edu/download/72329043/271.pdf

Rahmatullah, D. K., Nasution, S. M., & Azmi, F. (2016). Implementation of Low Interaction Web Server Honeypot Using Cubieboard. The 2016 International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC, 127-131. doi:10.1109/ICCEREC.2016.7814970.

Safa, N. S., & Solms, R. V. (2016). An information security knowledge sharing model in organizations. Computers in Human Behavior, 57, 442-451. doi:10.1016/j.chb.2015.12.037Purwanto, E. (2014, March 24). Keamanan Informasi. Retrieved from https://bpptik.kominfo.go.id/2014/03/24/404/keamanan-informasi/

Szczepaniuk, E. K., Szczepaniuk, H., Rokicki, T., & Klepacki, B. (2019). Information security assessment in public administration. Computers & Security, 90, 1-11. Retrieved from https://www.sciencedirect.com/science/article/pii/S0167404819302469

VIELBERTH, M., & PERNUL, G. (2018). A Security Information and Event Management Pattern. 12th Latin American Conference on Pattern Languages of Programs (SugarLoafPLoP 2018), 1-12. doi:10.5283/epub.41139

Yusuf, M. A., Wahono, B.Eng., M.Kom, M. M., Amanda, S. M., Putra, S. M., Williams, Ratna Dewi, S. M., . . . S.Tr.TP., J. (2018). Laporan Tahun 2018 Honeynet Project BSSN. Jakarta: Direktorat Deteksi Ancaman BSSN. Retrieved 2020, from https://bssn.go.id/laporan-tahunan-honeynet-project-bssn-ihp-2018/

Yusuf, M.T., A., Amanda, S.ST., M.M.Han., C. D., Putra, S.ST., M.M., I. A., Ratna Dewi, S.ST., M. A., Kartika Rachman, S.ST.TP., P. P., Yugitama, S.S.T., R., . . . Wahono, B.Eng., M.Kom., M. M. (2019). Laporan Tahunan Honeynet Project 2019 BSSN. Badan Siber dan Sandi negara, Direktorat Deteksi Ancaman BSSN. Jakarta Selatan: Badan Siber dan Sandi negara. Retrieved Desember 2020, from https://cloud.bssn.go.id/s/wz4ZYiYnWjRE6cb#pdfviewer