Deceptive directories and “vulnerable” logs: a honeypot study of the LDAP and log4j attack landscape

Shreyas Srinivasa
Aalborg University
Copenhagen, Denmark
shsr@es.aau.dk

Jens Myrup Pedersen
Aalborg University
Copenhagen, Denmark
jens@es.aau.dk

Emmanouil Vasilomanolakis
Aalborg University
Copenhagen, Denmark
emv@es.aau.dk

***Abstract*—The Lightweight Directory Access Protocol (LDAP) has been widely used to query directory services. It is mainly utilized for reading, writing, and searching directory services like the Active Directory. The vast adoption of LDAP for authentication has entailed several attack attempts like injection attacks and unauthorized access due to third-party key storage. Furthermore, recent vulnerabilities discovered in libraries like the *Log4j* can lead adversaries to obtain unauthorized information from the directory services through pivoting attacks. Moreover, the LDAP can be configured to operate on UDP, motivating adversaries to exploit it for Distributed Reflection Denial of Service attacks (DRDoS). This paper presents a study of attacks on the LDAP by deploying honeypots that simulate multiple profiles that support the LDAP service and correlating the attack datasets obtained from honeypots deployed by the *Honeynet Project* community. We observe a total of** $39,388$ **malicious events targeting the honeypots and discover** 273 **unique attack sources performing pivot attacks in a period of one month.**

***Index Terms*—LDAP, Honeypots, Deception, LDAP attacks**

1. Introduction

The Lightweight Directory Access Protocol (LDAP) has been used for querying and searching the directory services over many years. As the name suggests, LDAP is a lightweight implementation and the Internet variant of the Directory Assistance Service (DAS) from the X.500 protocol (aka. Directory Access Protocol) [1], [2]. Due to its light implementation, many applications support LDAP for synchronizing and managing directory services (e.g., the Active Directory Server from Microsoft). LDAP allows cross-platform clients to query the directory services that contain attribute-value pairs of users, applications, computers, and devices in the network through an LDAP client [3]. Enterprise applications use LDAP for authentication in applications that include email clients, SSH, server, and workstation access.

However, over the years, there have been many vulnerabilities in LDAP that enable injection attacks, unauthorized access, and remote code execution capabilities [4]–[6]. As many enterprise applications use LDAP for authentication, attackers are highly motivated to exploit the protocol to gain unauthorized access into the targeted infrastructure. According to the ENISA Threat Landscape Report 2021, there were several DDoS campaigns that leveraged UDP-based LDAP services in 2020. It was observed that a wave of DDoS attacks that targeted several Internet Service Providers in France, Belgium and Netherlands leveraged DNS and LDAP services for amplification attacks [7]. Furthermore, Internet scanning data from *Project Sonar* [8], shows up to three million LDAP services on the Internet with open TCP port 389 that accept unencrypted requests, implying that misconfigured LDAP services can lead to attacks of significant impact.

Honeypots are deception systems that simulate target systems or services. They work as decoys to attract attacks and store all the attack traffic. Traditionally, honeypots have been used to gather attacks from bots and as an effective source for threat intelligence data. There are several open-source honeypot projects, some maintained by the *Honeynet Project*, that are focusing either on specific protocols or vulnerabilities [9]. The simulation ranges across diverse application protocols used in IT, OT (Operational Technology), and IoT environments. Honeypots have been an obvious choice to study attack trends and, more recently, about attacker behavior psychology [10].

In this paper, we aim to extend and deploy a honeypot that simulates open-source implementations of directory services to gather attack trends in LDAP. Moreover, we add a *Log4j* component to our honeypots to allow an analysis of pivoting attacks towards LDAP. Furthermore, we enhance our findings by correlating them with attack data gathered from honeypots deployed by the Honeynet Project. We summarize our contributions as follows:

- We extend an open-source honeypot to simulate three different LDAP profile services.
- We deploy LDAP honeypots and perform an analysis of the attacks received on the honeypots.
- We correlate the attacks received in our honeypots with attack data from the Honeynet Project.

2. Related Work

In this section, we discuss related work in the areas of LDAP attack types and LDAP honeypots.

2.1. LDAP attacks

Several vulnerabilities have been reported on the LDAP over the years. These include Denial of Service attacks, remote code execution and privilege escalation on different independent LDAP implementations [11]. Furthermore, more recently, the LDAP has been exploited as a part of APTs that exploit other vulnerabilities (for example, CVE-2021-44228 of the Apache Log4j vulnerability) [12]. Early research from Alonso et al. show injection techniques possible through the LDAP [5]. The authors present injection techniques by manipulating the filters used for searching the directory services. Obimbo et al. present the risks of using LDAP as an authentication protocol by executing a DoS attack exploiting the TCP three-way handshake required for connection initialization with an LDAP server [4]. More recently, Jeitner et al. presented techniques to inject malicious payloads to launch injection attacks on protocols like DNS, LDAP, and Eduroam [6]. As LDAP is extensively used in enterprise infrastructure as an authentication service, any potential attack vector towards LDAP is of high risk.

2.2. LDAP honeypots

Early work on LDAP Honeypots was proposed from Grimes [13]. The author provides an overview of honeypots in general and Windows-based honeypots that administrators can deploy to detect potential zero-day attacks. Furthermore, the author provides an overview for modeling honeypots for windows-based environments and protocols by using scripts from the *HoneyD* honeypot framework [14]–[16]. The *HoneyD* honeypot framework acts as a daemon that can create virtual hosts on a network that can be configured to run arbitrary services. The daemon can run on multiple addresses and provide scripts to emulate an entire device or a specific protocol. Moreover, there is active research that proposes using Honeytokens, a subset of honeypots that emulate a digital entity like user accounts, files, and folders to detect malicious activity or infections. For instance, Lukas et al. propose the creation of fake user accounts as honeytokens on Active Directory Server to capture malicious access attempts [17].

The T-Pot project [18] is a collection of 25 different honeypots that includes the Log4Pot honeypot [19]. Log4Pot simulates a vulnerable Log4j environment and can be configured to listen on multiple ports. The honeypot further provides a log analysis tool that extracts the attack payloads, decodes them and builds a timeline of attacks. The GreedyBear Project [20] aggregates the attack data from the honeypots of the T-Pot project, specifically from the Log4Pot and Cowrie honeypots, and converts them into actionable feeds to facilitate threat intelligence. The GreedyBear project is currently maintained by the Honeynet Project [9] and provides public access to feeds aggregated by the GreedyBear project. Nevertheless, there is no work on honeypots that aims at capturing attacks specific to LDAP. We address this gap by extending an open-source honeypot to simulate directory services with LDAP and capture the attacks [21].

3. Methodology

This section presents the methodology for the LDAP honeypot implementation, the experimental setup and the analysis of attack data from the *Honeynet Project* community.

3.1. LDAP honeypot

To simulate LDAP service, we extend RIoTPot, an open-source honeypot that is modular and capable of operating in hybrid-interaction levels [21]. RIoTPot provides high-interaction capability by running services on dedicated, ephemeral containers with capturing the traffic as *pcap* files and in an attack database. Leveraging the modular feature of RIoTPot, which facilitates easy integration of protocols and services into the simulation portfolio, we integrate three profiles: Apache Directory Server [22], OpenLDAP [23] and OpenDJ [24]; that support the LDAP service and run them in containerized mode. We set up individual containers of the three profiles and utilize RIoTPot's orchestration and logging features to capture the attack traffic. Furthermore, we simulate a webservice with the *Log4J* vulnerability [12] that refer to the directory services simulated by the profiles in containers. In total, we deploy three webservices that connect to individual directory services. We describe the simulated profiles in detail below.

3.1.1. Apache Directory Service. The Apache Directory Server (ADS) [22] is an open-source, extendable implementation of Directory services. The service is implemented using the Java programming language and can be embedded as a module in a server application. ADS supports the communication through LDAP and is compliant with the LDAP v3. In addition to the LDAP, ADS supports Kerberos 5 and the Change Password protocols. Furthermore, ADS uses an adaptation of the X.500 basic access control scheme with subentries to control access and attributes within the Directory Information Tree (DIT). The directory service can be configured through an LDIF file, a known format to define the properties of DIT, directory objects, and attributes. The Apache community actively maintains the ADS open-source repository.

3.1.2. OpenLDAP. OpenLDAP is an open-source implementation of LDAP [23]. The package includes a standalone LDAP load-balancing daemon (*lloadd*) , a standalone LDAP service daemon (*slapd*) and libraries that implement LDAP with additional utilities. The *lloadd* listens for LDAP connections on a specified number of ports and forwards the LDAP operations received over these connections to be processed by the backend, while the *slapd* listens to incoming LDAP requests and responds to the LDAP queries received over the connections. In addition, the *slapd* offers operation in *tool* mode which provides multiple profiles for the daemon.

3.1.3. OpenDJ. OpenDJ is an opensource LDAPv3 compliant implementation of the directory service, developed using Java [24]. The implementation features scalability for large domains, monitoring tools, and replication between multiple instances. In addition to LDAP v3, OpenDJ supports the Directory Service Markup Language (DSMLv2). The OpenIdentity Platform actively maintains the OpenDJ project.

3.1.4. HTTP Service with Log4j vulnerability. Log4j is an open-source logging Java library that provides multiple logging levels for debugging applications. The library is extensively used by applications developed in Java. Recently, a bug in the Log4j library was disclosed in which an attacker can perform remote code execution on the victim using the library for debug-logging [12]. This vulnerability allows unauthorized users to run arbitrary code on the target machine when a configuration uses a JDBC Appender with a JNDI LDAP data source URI [25]. Attackers can spawn malicious LDAP servers to carry out the Log4j attacks on the victims. To understand if there are any potential pivot attacks, that may target the LDAP services through the Log4j exploit, we enhance our honeypot instances (see also experimental setup below) with an HTTP service that showcases the Log4j vulnerability and configure them to connect to individual directory services. The websites simulate a login dashboard with a welcome header, fields for user login, and a login button. The login button performs a standard procedure of verifying the username and password from the directory service configured. The websites are each hosted on the same instance as the directory simulations, and a search user is configured with the websites to be able to search the directories, which enables the examination of LDAP injection attacks.

3.2. Experimental setup

To capture attacks on individual profiles, we deploy RIoTPot on three hosts, with each RIoTPot instance simulating a directory service and an HTTP service. Figure 1 shows the experimental setup of the honeypots in our lab environment. Each host is assigned a public IP address and has ports 389 (LDAP) and 80 (HTTP) open to the Internet. The traffic from each host is captured as a pcap file and stored in a remote file repository. Furthermore, all traffic received on ports 80 and 389 are logged in an attack database. The file repository and the attack database are set up on a remote host to avoid disruption in logging in case of a crash. The directory service is configured with basic authentication and is set with an admin username with a non-complex password. We configure all the directory services with the same domain name (LDAP.xxx.xx) and are initialized with five organization units and 120 users to look similar to a production service.

3.3. Honeynet Project dataset

To get a holistic view of attacks, we analyze the data from the honeypots deployed by the Honeynet Project community. In particular, we request the feed from the GreedyBear [20] project that aggregates attacks towards the Log4j vulnerability. We correlate these logs to the findings of our own honeypots. Upon analysis of the Honeynet Project data, we identify JNDI calls in the payloads and find similar attacks in our honeypots. We describe our findings in Section 5.

4. Results

This section lists our findings on the attacks gathered from our honeypots.

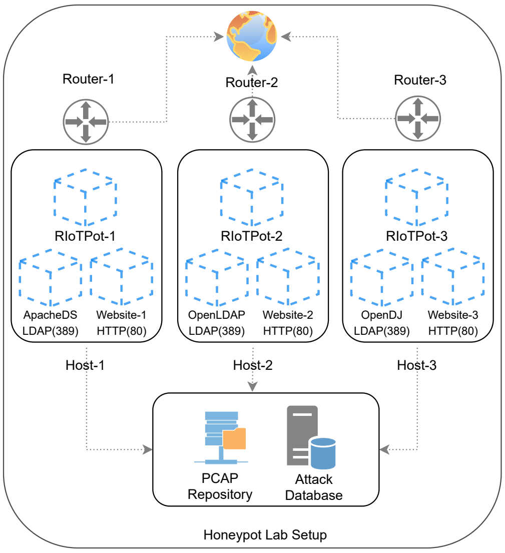

Figure 1. Overview of our experimental setup

4.1. Attack traffic count

We deploy three profiles of open-source Directory Services that support LDAP and add three vulnerable websites with Log4j vulnerability associated with each profile. We classify suspicious traffic as an LDAP attack when an injection pattern or an irregular search is observed in the traffic [5]. Similarly, on the HTTP, we classify the traffic as an attack when brute-force attempts and remote code execution patterns are detected. Figure 2 summarizes the number of attacks received on each directory service profile on ports 389 and 80 for 30 days. At a glance, we received at total of $39,388$ attacks. The OpenLDAP directory service received the highest number of attacks on LDAP (2613) in comparison to ApacheDS (2414) and OpenDJ (2341). We observe that the attacks increased after the first 14 days of the deployment on all three profiles. We suspect this could be because of possible listing on the Internet-wide scanning services. Note that the attacks shown are exclusive of probing traffic from known Internet-scanning services. In particular, the HTTP service received a total of 22,673 events and the LDAP received 8,100 events from known scanning services. The traffic from these benign scanning services was identified using the noise-filter module of RIoTPot [21].

4.2. Attack sources

As a result of exposing our honeypots to the Internet, we receive high traffic volume, primarily benign, from Internet-wide scanning services. Figure 3 shows the distribution of traffic from scanning services (benign) and attack traffic with malicious intent. RIoTPot filters the traffic received on the honeypots by identifying the probing traffic from 19 Internet-wide scanning services [21]. Filtering of benign scanning traffic reduces the noise in the gathered data, thereby concentrating on the remaining suspicious traffic. All traffic towards the honeypot instances can be considered suspicious as there is no productive value in interaction with a honeypot. We label suspicious traffic to be an attack upon observing malicious intent in the

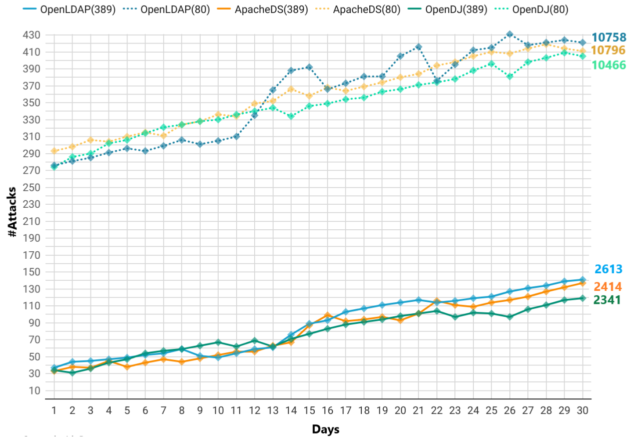

Figure 2. Attacks received over 30 days - LDAP and HTTP

requests. We observe that the OpenLDAP profile received the highest number of malicious requests compared to the other profiles. The honeypots received traffic from 273 unique attack sources.

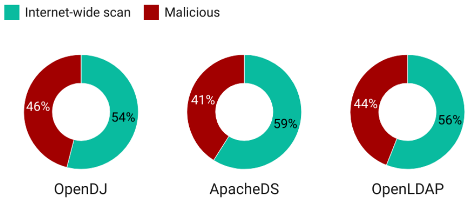

Figure 3. Traffic classification on honeypots

4.3. Attack types

We observe multiple attack types in our honeypots, including many LDAP injection attacks, suspicious search, remote code execution, and brute-force attempts. Figure 4 shows the percentage of different attack types received on each simulated directory service.

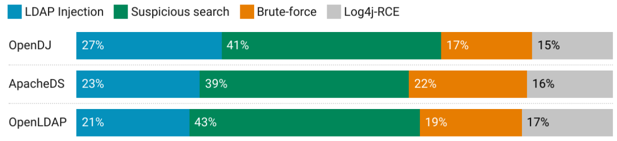

Figure 4. Attack types received on honeypots

The OpenDJ profile received the most LDAP Injection attacks in comparison to the other profiles. The attacks aimed at bypassing the authentication by using blind exploitation techniques to fetch the *userPassword* attribute. The profiles further received random suspicious search queries with logical-operators on the LDAP filters. Moreover, we identified many brute-force attempts on the HTTP webservice. In addition to the brute-force attacks, the websites received attacks that exploited the Log4j vulnerability. We observe fewer attacks towards Log4j in comparison to the other attack types and this could be because of the time elapsed since the disclosure of the vulnerability.

5. Discussion

In this section, we discuss our findings from the analysis of the attack data received from the *Honeynet* community and additional findings from the attack data received on our honeypots.

5.1. Correlating data from the Honeynet Project

The data obtained from the Honeynet Project is an aggregated feed from GreedyBear [20]. The project aggregates data from 30 Log4j honeypot instances. First, we correlate the attack sources observed on both datasets. Over a period of 30 days, the GreedyBear feed had an average of 3,269 events per day and 693 unique source IPs. Figure 5 shows the correlation of the number of unique IPs that have been observed on Honeynet data and our honeypots over the same period of 30 days. The number of same actors denote the total attack sources observed on both honeypot datasets and the different actors denote the attack sources that were observed exclusively on our honeypots. Upon further analysis, we find that the different actors observed on our honeypots targeted also the LDAP service. The different actors observed on our honeypots may be the result of running both LDAP and Log4j simulations. The attack sources shown in the figure include the attacks received only on the Log4j simulation in our honeypots. Furthermore, we find recurring probes from attack sources that are not from known Internet-wide scanning services and appear to be performing pivot attacks. In addition, we examined the code that was called through RMI to find patterns. Upon analysis we find similarities in the code that aimed at performing LDAP injections from many sources.

5.2. Attack samples

We list sample attacks in appendix Table 1 for each attack type categorized in Figure 4. The table further lists different LDAP injection attack types and samples observed on our honeypots. The *Authentication Bypass* attacks aimed at injecting filtered LDAP queries with sequences to bypass authentication. The privilege escalation attacks aim at listing unauthorized directory contents bypassing a search sequence with a low-security level. We observe blind injection attacks that request a Boolean operation to check if an *admin* class exists that belongs to a *domain* type. In addition, the honeypot instances received many suspicious search query requests. For instance, the sample listed in Table 1 requested a sequence from the LDAP service on the same host. This search entails that the adversary previously performed reconnaissance to discover open LDAP ports on the host. Many brute force attacks were identified in which adversaries tried to log in via a list of passwords. We further determine, by checking the word list order, that the passwords used were part of the NMap default password list [26]. Lastly, there were Log4j attacks observed that performed RMI calls. We list some sample Log4j exploits received in Table 1.

5.3. Pivot attacks

Pivoting attacks can be described as attacker movement from one compromised system to more systems

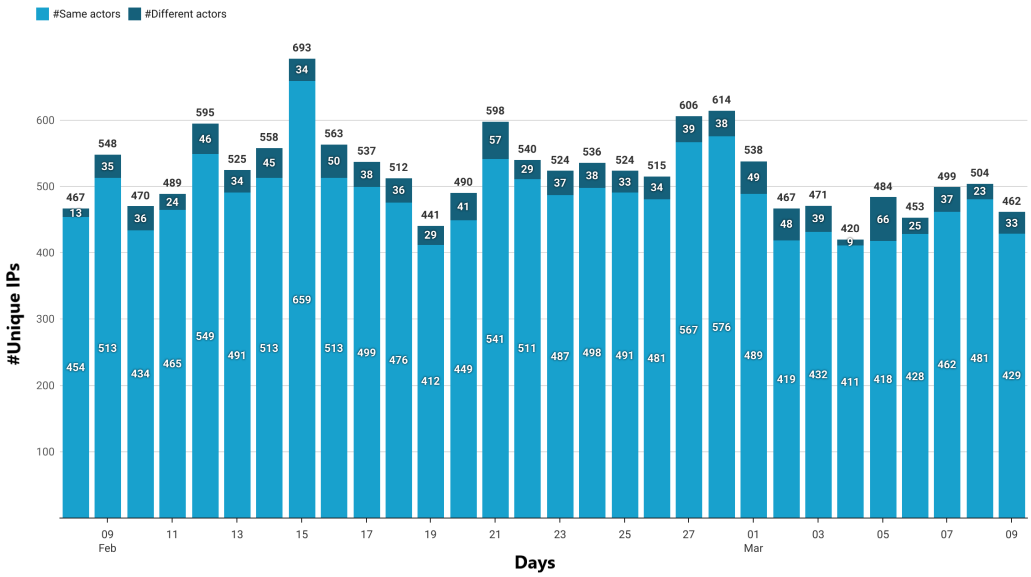

Figure 5. Correlation of attack sources from the Honeynet Project and our honeypots

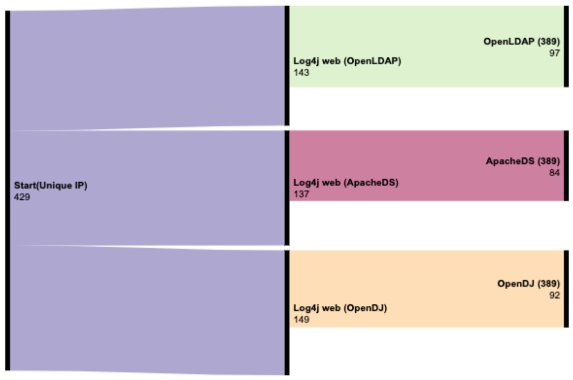

Figure 6. Pivot attacks overview

within the same or remote infrastructure. We observe some attacks that try to pivot into the directory services by leveraging the Log4j vulnerability through LDAP injection techniques. Upon examining the code from RMI calls specified through JNDI, we find LDAP filters that aim to list all organizational units and enumerate domain users and domain admins groups. We observe such attacks on all three simulation profiles of our experiment. Figure 6 depicts the number of pivoting attacks observed on each simulation profile. The attacks begin with targeting the Log4j vulnerability, and sequentially move on to target the simulated directory services through LDAP. We observe that out of 429 unique attack sources (observed exclusively on Log4j), 273 of them attempted pivot attacks on the directory services.

5.4. Limitations

We acknowledge the following limitations in our approach. First, we exclusively consider open-source implementations of directory services and LDAP. This limits our scope as most enterprises use Microsoft Active Directory as their directory service [27]. Second, our work is further limited in the simulation of LDAP operational modes, such as LDAPS and CLDAP. The simulation of CLDAP would provide an overview of the reflection-based attacks. Third, though we simulate a high-interaction profile for the directory services and LDAP, we limit the experiment in terms of the domain simulation by using an unregistered domain. Hence, using a registered domain in our experiment may enhance the deception layer and appear more attractive for adversaries. Lastly, the total attack events observed on each profile are the result of a month study only; an extended study is needed for a more holistic understanding of the field.

5.5. Ethical considerations

As honeypots are systems that simulate vulnerable environments, they can be leveraged by adversaries to cause attacks on the Internet. To prevent such attacks, we limit the egress traffic from our honeypots. Furthermore, the containers spawned from our honeypots for simulation are ephemeral, such that new instances are created periodically to avoid spread of infections. In regards to the dataset from the Honeynet project, we take care in not disclosing the IP addresses of honeypots deployed by the community.

6. Conclusion

This paper conducts a honeypot study of the attacks on LDAP by deploying three open-source directory service profiles with the webservers simulating the Log4j vulnerability. We observe many attack types, including LDAP injection attacks and suspicious search queries. Lastly, we summarize the attack types and correlate our findings with the data from the Honeynet community. As future work, we aim to perform a longitudinal study of LDAP honeypots with extended profiles that include the Active Directory.

Acknowledgement

This research was supported as part of COM^3, an Interreg project supported by the North Sea Programme of the European Regional Development Fund of the European Union. We would like to thank Matteo Lodi from the Honeynet Project for sharing the dataset. We would like to thank Reinholdt W. Jorck og Hustrus Fond and Otto Mønsteds Fond for facilitating the research. Lastly we thank the reviewers for their valuable comments.

References

[1] M. Rose, "Directory assistance service," in *RFC 1202, Performance Systems International, Inc.* Citeseer, 1991.

[2] B. Smetaniuk, "Distributed operation of the x. 500 directory," *Computer Networks and ISDN Systems*, vol. 21, no. 1, pp. 17–40, 1991.

[3] M. Wahl, T. Howes, and S. Kille, "Rfc2251: Lightweight directory access protocol (v3)," 1997.

[4] C. Obimbo, B. Ferriman *et al.*, "Vulnerabilities of ldap as an authentication service." *J. Information Security*, vol. 2, no. 4, pp. 151–157, 2011.

[5] J. M. Alonso, R. Bordon, M. Beltran, and A. Guzmán, "Ldap injection techniques," in *11th IEEE Singapore International Conference on Communication Systems*. IEEE, 2008, pp. 980–986.

[6] P. Jeitner and H. Shulman, "Injection attacks reloaded: Tunnelling malicious payloads over dns," in *30th USENIX Security Symposium (USENIX Security 21)*, 2021, pp. 3165–3182.

[7] A. Claudio, C. Stephen, S. Andreas, and D. Christos. (2021) Enisa threat landscape report 2021. [Online]. Available: https://www.enisa.europa.eu/publications/enisa-threat-landscape-2021/@@download/fullReport

[8] Rapid7, "Project sonar," 2021. [Online]. Available: https://opendata.rapid7.com/sonar.tcp/2021-12-01-1638342851-tcp_ldap_389.csv.gz

[9] T. H. Project. (2021) The honeynet project. [Online]. Available: https://www.honeynet.org/

[10] K. J. Ferguson-Walter, M. M. Major, C. K. Johnson, and D. H. Muhleman, "Examining the efficacy of decoy-based and psychological cyber deception," in *30th USENIX Security Symposium (USENIX Security 21)*, 2021, pp. 1127–1144.

[11] MITRE. (2021) Ldap vulnerabilities and disclosures. [Online]. Available: https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=ldap

[12] A. S. Foundation. (2021) Cve-2021-44228. [Online]. Available: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2022-23302

[13] Springer, Ed., *Windows Honeypot Modeling*. Berkeley, CA: Apress, 2005, pp. 63–88. [Online]. Available: https://doi.org/10.1007/978-1-4302-0007-9_3

[14] N. Provos, "Honeyd-a virtual honeypot daemon," in *10th DFN-CERT Workshop, Hamburg, Germany*, vol. 2, 2003, p. 4.

[15] N. Provos and T. Holz, *Virtual honeypots: from botnet tracking to intrusion detection*. Pearson Education, 2007.

[16] N. Provos *et al.*, "A virtual honeypot framework." in *USENIX Security Symposium*, vol. 173, no. 2004, 2004, pp. 1–14.

[17] O. Lukas and S. Garcia, "Deep generative models to extend active directory graphs with honeypot users," *arXiv preprint arXiv:2109.06180*, 2021.

[18] T. Security, "T-pot - the all in one honeypot platform," 2022. [Online]. Available: https://github.com/telekom-security/tpotce

[19] P. Thomas, "A honeypot for the log4shell vulnerability (cve-2021-44228)," 2022. [Online]. Available: https://github.com/thomaspatzke/Log4Pot

[20] (2022) Greedybear honeypot feed. Honeynet Project. [Online]. Available: https://github.com/honeynet/GreedyBear

[21] S. Srinivasa, J. M. Pedersen, and E. Vasilomanolakis, "Riotpot: a modular hybrid-interaction iot/ot honeypot," in *26th European Symposium on Research in Computer Security (ESORICS) 2021*. Springer, 2021.

[22] Apache. (2021) Apache directory. [Online]. Available: https://directory.apache.org/apacheds/

[23] O. Kuzník. (2021) Openldap. [Online]. Available: https://www.openldap.org/

[24] OpenIdentityPlatform. (2021) Opendj. [Online]. Available: https://www.openidentityplatform.org/opendj

[25] (2021) Cve-2021-44832. Apache Software Foundation. [Online]. Available: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2021-44832

[26] G. Lyon, "Nmap network mapper," 2021. [Online]. Available: https://nmap.org/

[27] S. Reimer and M. Mulcare, *Active Directory® for Microsoft® Windows® Server 2003 Technical Reference*. O'Reilly Media, Inc, 2009.

Appendix A. Samples of attack types

Table 1 lists the sample attacks received on our honeypots like LDAP injection, suspicious search queries, brute-force attacks and the Log4j RMI attacks. The table further lists the different types of LDAP injection attacks in particular the authentication bypass technique which aims to gain unauthorized access by injection of a filter that ignores the password attribute in the LDAP query, the privilege escalation attacks which aims at fetching unauthorized information and blind injection attacks that aims at fetching boolean information about specific objects in the directory.

Attack-type	**Received Attack Sample**
LDAP-Injection Authentication Bypass	&(USER=admin)(&))(PASSWORD=Pwd)
LDAP -Injection Privilege elevation	"www)(security_level=*))(&(directory=html"
LDAP -Injection Blind LDAP Injections	(&(objectClass=admin*)(type=domain*))
Suspicious search	GET /?x=$jndi:ldap://127.0.0.1
Brute-force	#cn=root,cn=users,dc=resilient,dc=dk password
Log4j-RCE	GET /$%7Bjndi:$%7Blower:l%7D$%7Blower:d%7Da$%7Blower:p%7D://************.*.psc****

TABLE 1. SAMPLES OF ATTACKS RECEIVED ON HONEYPOTS