PenTest: `website.local`

Andreas Happe

Inhaltsverzeichnis

Kapitel 1

Executive Summary (max. eine Seite)

In „einfacher" Sprache sollte der Inhalt des Berichts zusammengefasst werden. Ziel dieses Textes sind nicht Techniker, sondern z.B. Management. Nach Lesen des Executive Summaries sollte dieses den Umfang und die Ergebnisse bzw. die Auswirkungen des Penetration-Tests verstehen.

- Was wurde getestet?
- Welche Ergebnisse konnten vorgefunden werden?
- Was sind die potentiellen Auswirkungen? Welchen Einfluss hat das Ergebnis auf den Betrieb der Webseite?
- Empfehlungen?

Kapitel 2

Die getestete Applikation (ca. eine Seite)

Hier sollte die getestete Seite kurz beschrieben werden. Falls der Bericht nach mehreren Monaten wieder gelesen wird, ist das ursprüngliche Testziel eventuell nicht mehr nachvollziehbar (URLs können sich auch ändern) bzw. sollte auch die Testumgebung kurz erläutert werden.

1. Welche Applikation wurde getestet? Was ist deren Aufgabe?
2. Welche Gefährdungen werden gesehen? Vor was hat der Kunde Angst (eigene Annahmen)
3. Wer sind die potentielle Angreifer?
4. Beschreibung des Ablaufs. Gab es eine Produktiv- oder Test-Umgebung? Durfte destruktiv getestet werden? Gab es Bereiche die nicht getestet werden durften?

Kapitel 3

Schwachstellen und Verbesserungsmassnahmen

Hier sollten die vorgefundenen Schwachstellen zusammengefasst werden. Ebenso sollte hier eine Übersicht über die vorgeschlagenen Verbesserungsmassnahmen gegeben werden. Dies ist quasi das Gegenstück zur Executive Summary für „Techniker"

- Welche Schwachstellen wurden vorgefunden?
- Welche Schwachstellen werden besonders kritisch befunden? Eventuell Sortierung der Schwachstellen nach Kritikalität? Tabellen, etc. können hier gerne verwendet werden
- Wie können diese behoben werden?
- Gibt es weitere empfohlene Absicherungsmassnahmen (Hardening)?

Kapitel 4

Datensammlung

Dieses Kapitel sollte den Test nachvollziehbar machen. Ziel der Datensammlung ist der „Beweis" für die vorgefundenen Schwachstellen. Aufgrund der Datensammlung sollte der Kunde fähig sein, die Schwachstellen selbst nachvollziehen zu können. Ebenso kann hier auf verschiedene Gegenmassnahmen eingegangen werden (von denen im Kapitel „Schwachstellen und Verbesserungsmassnahmen" eine empfohlen werden sollte). Für die verschiedenen Bereiche der Datensammlung werden Subkapitel empfohlen.

4.1 Verbindungssicherheit

Was kann zur Verbindungssicherheit gesagt werden?

- HTTPS/TLS Konfiguration
- wurden Massnahmen zur automatischen Gewährleistung der Verbindungssicherheit (HSTS, CSP, etc.) gesetzt?
- kann z.B. mittels *testssl.sh* oder *Qualys SSLTest* getestet werden

4.2 Verwendete HTTP-Header

Am besten Ausgabe der verwendeten HTTP-Header (z.b. auf der Startseite oder innerhalb des eingeloggten Bereichs):

```
HTTP/2 200 OK
date: Fri, 19 Feb 2021 15:05:19 GMT
content-type: text/html
last-modified: Tue, 16 Feb 2021 09:38:09 GMT
```

```
vary: Accept-Encoding
strict-transport-security: max-age=15552000; includeSubDomains
x-content-type-options: nosniff
server: cloudflare
content-encoding: br
```

Anschließend Diskussion der vorgefundenen HTTP-Header. Welche sicherheitsrelevanten Header wurden vorgefunden, welche haben gefehlt. Welche Empfehlungen gibt es? Siehe auch Mozilla Observatory oder securityheaders.io

4.3 Software-Enumeration

Dieses Kapitel soll die vorgefundenen Softwarekomponenten erwähnen. Hier sollte überprüft werden, ob diese auf dem letzten sicherheitsrelevanten Patch-Level sind bzw. ob es bekannte Exploits für diese Softwarekomponenten bekannt sind.

- Welche Software wurde in welcher Version vorgefunden?
- Gibt es bekannte Schwachstellen?
- Wurden weitere Artefakte (*.git*, Admin-Tools, Backup-Files) vorgefunden?

4.4 Session, Login/Logout, Authentication

Dieses Kapitel sollte Fragen zum Thema Benutzerverwaltung bzw. Benutzersessions beleuchten.

- Wie werden Benutzersessions abgebildet? Wie wurden diese abgesichert? Schwachstellen und Verbesserungsmassnahmen?
- Gibt es Auffälligkeiten bei Login/Logout?
- Falls Tokens verwendet werden? Wie sind diese aufgebaut? Gibt es hier Probleme?
- Kann man auf Ressourcen ohne Login zugreifen?

4.5 Berechtigungskonzept

Dieses Kapitel sollte das vorgefundene Berechtigungskonzept genauer erläutern. Es sollte auch (stichprobenweise) getestet werden, ob das Zugriffskonzept auch implementiert wurde (ob Benutzer einer Gruppe wirklich nur auf die Daten und Operationen einer Gruppe zugreifen können. Falls es sich um ein „friendly" Opfer handelt, kann hier auch um einen Administrator-Account gefragt werden. Dieser dient jetzt nicht für den Test direkt, sondern wird verwendet um mögliche Admin-Operationen zu identifizieren auf die dann, als normaler Benutzer, versucht wird zuzugreifen.

- Kann ich auf Daten anderer Benutzer zugreifen?
- Kann ich das Profil eines anderen Benutzers modifizieren?

4.6 Injection-Angriffe

Sammelkapitel für einzelne Injection-Angriffe. Initial sollte bestimmt werden, welche Angriffsvektoren für die getestete Applikation sinnvoll erschienen. So wird z.B. eine LDAP-Injection wahrscheinlich unrealistisch bei einem eCommerce-Shop sein, ebenso wird eine SQL-Injection primär bei einem System mit einem Datenbank-Backend vorkommen. Potentiell können die Angriffe weiters in Client- und Server-Seitige Angriffe aufgeteilt werden.

Typische Fragen:

- Gibt es verwundbare Operationen?
- Wie wurden diese getestet?
- Falls Schwachstellen gefunden wurden, wie können diese ausgebessert werden?

4.6.1 Serverseitig

...

4.6.2 Clientseitig

...