|  |
| --- |
|    Dvořákovo nábrežie 10811 02 BratislavaSlovakiaP.O. Box 7820 04 Bratislava 24SlovakiaTelephone +421 2 59984 111Telefax +421 2 59984 333Internet www.kpmg.sk KPMG Slovensko Advisory, k.s.KPMG Slovensko Advisory, k.s. KPMG Slovensko Advisory, k.s., the Slovak member firm of KPMG global organization of independent member firms affiliated with KPMG International Limited.All rights reserved. Február 2023Záverečná správaExterný infraštruktúrny a aplikačný penetračný test Customer, s.r.o. |
| Príloha č.1 |
| Príklad správy |

Kontrola a schválenie dokumentu

História revízií

|  |  |  |  |
| --- | --- | --- | --- |
| Verzia | Autor | Dátum | Revízia |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Tento dokument skontroloval

|  |  |  |
| --- | --- | --- |
|  | Revidoval | Dátum revízie |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Tento dokument schválil

|  |  |  |
| --- | --- | --- |
|  | Odborníci na danú problematiku |  |
|  | Meno | Podpis | Dátum revízie |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Obsah

1 Úvod 1

1.1 Ciele 1

1.2 Rozsah 1

1.3 Náš prístup 1

1.4 Formát správy 3

1.5 Obmedzenia 3

1.6 Dôvernosť 3

2 Zhrnutie pre manažment 5

2.1 Rozsah prác 5

2.2 Hlavné zistenia 5

3 Externý infraštruktúrny penetračný test 7

3.1 Vybrané systémy 7

3.1.1 SSH Server podporuje algoritmy v CBC móde 7

3.1.2 Služba SSL/TLS používa na zabezpečenie svojej komunikácie samo-podpísaný certifikát 8

3.1.3 Povolenie nešifrovaných spojení s POP3 serverom 9

4 Externý aplikačný penetračný test (mcx.Zákazník.sk) 11

4.1 Aplikačná infraštruktúra 11

4.1.1 Využívanie zraniteľnej JavaScript knižnice (AngularJS) 11

4.2 Autentifikácia 11

4.2.1 Možnosť hádania prihlasovacích mien cez funkciu pre reset hesla 11

4.3 Správa relácie (Session Management) 12

4.3.1 Identifikátor relácie prenášaný v URL 12

4.4 Validácia vstupov 13

4.4.1 Možnosť vloženia dát vo formáte JSON na strane klienta 13

# Úvod

Táto správa sumarizuje výsledky externého infraštruktúrneho a aplikačného penetračného testu vybraných systémov spoločnosti Zákazník, s.r.o. (ďalej ako “Zákazník” alebo „spoločnosť“). Testovanie bolo vykonané v období od XXX do XXX. Re-testovanie na prítomnosť zraniteľnosti Log4Shell prebiehalo v období od XXX do XXX.

## Ciele

Zákazník si vybral POSKYTOVATEĽ, s.r.o. za účelom vykonania externého infraštruktúrneho a aplikačného penetračného testu vybraných systémov spoločnosti. Hlavným cieľom posúdenia bezpečnosti bolo identifikovanie zraniteľností na úrovni infraštruktúry a aplikácie, ktorých zneužitie útočníkom by mohlo viesť k úspešnému útoku na dôvernosť, integritu alebo dostupnosť systémov a tým aj nimi spracovávaných údajov a procesov.

## Rozsah

Spoločnosťou bol odsúhlasený nasledovný rozsah prác:

* Externý infraštruktúrny penetračný test,
	+ IP1
	+ IP2
	+ IP3
	+ Web1.sk
	+ Web2.sk
* Externý aplikačný penetračný test,
	+ Web1.sk

## Náš prístup

Externý infraštruktúrny a aplikačný penetračný test vybraných systémov spoločnosti bol založený na POSKYTOVATEĽ metodológii pre penetračné testy.

Táto metodológia definuje nasledovné fázy:

* Fáza plánovania a prípravy. Počas tejto fázy sa so Spoločnosťou odsúhlasí rozsah, prístup a časový rozvrh testov.
* Fáza testovania infraštruktúry. V tomto kroku sa vykoná mapovanie dostupných sieťových služieb, detailné testovanie každej jednej služby a vyhodnotenie synergetického efektu identifikovaných zraniteľností.
* Fáza testovania aplikácie. V tomto kroku sa vykoná prieskum aplikácie s cieľom zmapovania jej funkcií, štruktúry a technológií na ktorých je vybudovaná. Následne sú dostupné funkcie aplikácie detailne pretestované vo všetkých relevantných oblastiach metodológie OWASP a vyhodnotený celkový efekt identifikovaných zraniteľností.
* Fáza reportovania. Výsledky testov sú sumarizované v záverečnej správe. Relevantné výstupy sú uvedené v prílohe.

Na základe vyššie spomenutej metodológie bola vykonaný penetračný test v nasledovných krokoch:

* Fáza plánovania a prípravy,
* Fáza testovania infraštruktúry,
* Fáza testovania aplikácie,
* Fáza reportovania.

Penetračný test infraštruktúry bol vykonaný na základe metodiky OSSTMM. Testy pozostávali z cyklického vykonávania nasledujúcich fáz:

* Attack surface discovery – pri dohodnutom rozsahu prác išlo o mapovanie všetkých dostupných otvorených sieťových portov na cieľových systémoch
* Identifikácia sieťových služieb
* Testovanie konfigurácie služieb a enumerácia služieb
* Analýza a zneužitie zraniteľností

Cieľové systémy boli testované aj na prítomnosť zraniteľnosti Log4Shell.

Penetračný test aplikácie bol vykonaný na základe metodiky OWASP. Testy sa vykonávali len z pohľadu neautentifikovaného testera a pozostávali najmä z nasledujúcich fáz:

* Zhromažďovanie informácií,
* Testovanie autentifikácie,
* Testovanie autorizácie,
* Testovanie riadenia relácie (session management)
* Testovanie validácie vstupov,
* Testovanie spracovania chýb,
* Testovanie možného použitia slabej kryptografie

## Formát správy

Zistenia sú zoradené do logických kategórií. Rizikovosť jednotlivých zistení je uvedená s nasledovnými úrovňami:

**Vysoké** – priama hrozba, je potrebné ju okamžite riešiť.

**Stredné** – nepriama hrozba, akcia by mala byť založená na ohodnotení úrovne rizika.

**Nízke** – nepotrebná funkcionalita alebo nastavenie, jednoduchá korekcia.

## Obmedzenia

Vykonané penetračné testovanie podliehalo nasledovným obmedzeniam:

* + - Testovanie bolo vykonané z pohľadu neautentifikovaného testera, ktorý nemal k testovaným systémom nadštandardný prístup na sieťovej, systémovej ani aplikačnej úrovni
		- Testeri (resp. ich IP adresy) neboli počas penetračneho testovania umiestnené na tzv. white-list bezpečnostných systémov (firewally, WAF, IPS, AV, ...) na perimetri siete Zákazníka, takže tieto systémy mohli ovplyvniť presnosť testov
		- Testovanie web aplikácie na URL portal.Zákazník.sk bolo vykonané z pozície neautentifikovaného testera, takže testovaná bola len verejne prístupná časť aplikácie.

## Dôvernosť

Táto správa je vypracovaná za účelom informovania manažmentu spoločnosti. Citácia tejto správy alebo jej časti, ako aj odvolávanie sa na ňu nie je možné bez predchádzajúceho písomného súhlasu POSKYTOVATEĽ. POSKYTOVATEĽ nie je oprávnená poskytovať tretím stranám akékoľvek dôverné informácie obsiahnuté v tejto správe bez predchádzajúceho písomného súhlasu spoločnosti.

# Zhrnutie pre manažment

## Rozsah prác

POSKYTOVATEĽ vykonal pre Zákazníka nasledovné práce:

* Externý infraštruktúrny penetračný test,
* Externý aplikačný penetračný test;

Nasledujúca tabuľka sumarizuje testované systémy aj s informáciou či bolo na konkrétnom systéme identifikované zistenie:

|  |  |
| --- | --- |
| Cieľový systém | Boli identifikované zistenia |
| IP1 | Nie |

## Hlavné zistenia

**Externý infraštruktúrny penetračný test**

Bolo zistené, že SSL/TLS služba pre poštový IMAP server, používa na zabezpečenie svojej komunikácie digitálny certifikát, ktorého platnosť vypršala v máji 2018.

Taktiež bolo zistené, že SSL/TLS služba pre rôzne poštové servery používa na zabezpečenie svojej komunikácie samo-podpísaný (self-signed) certifikát.

Použitie exspirovaných a samo-podpísaných certifikátov výrazne ohrozuje bezpečnosť prenášaných údajov, ktorými sú v tomto prípade prihlasovacie údaje a obsah e-mailovej komunikácie.

Ďalej bolo zistené, že jeden z poštových serverov POP3 umožňuje pripojenia klientov použitím nešifrovaného spojenia.

Nezabezpečené sieťové protokoly prenášajú údaje v otvorenom tvare. To znamená, že ktokoľvek s prístupom k takejto sieťovej komunikácii má možnosť použitím voľne dostupných softvérových nástrojov sledovať, prípadne aj modifikovať prenášané údaje. To môže viesť k úniku citlivých údajov prípadne inému spôsobu narušenia bezpečnosti v dôsledku manipulácie prenášaných údajov.

**Externý aplikačný penetračný test**

Funkcia pre reset hesla tiež umožňuje efektívne hádať platné prihlasovacie mená, keďže vracia rôznu odpoveď pre existujúce prihlasovacie meno a pre neexistujúce prihlasovacie meno.

Identifikované prihlasovacie mená môžu byť následne použité pre hádanie hesiel cez prihlasovací formulár aplikácie.

# Externý infraštruktúrny penetračný test

## Vybrané systémy

### SSH Server podporuje algoritmy v CBC móde

| Riziko | ID zistenia | Zistenie |
| --- | --- | --- |
| Nízke | 3.1.1 | SSH Server podporuje algoritmy v CBC móde |

##### Zistenie

Bolo zistené, že SSH server je nakonfigurovaný tak, že podporuje spojenia šifrované algoritmami v CBC (Cipher Block Chaining ) móde.

Zraniteľný systém:

* abc.Zákazník.sk, port 1 (TCP)

Pre spojenia client-to-server sú povolené nasledovné Cipher Block Chaining (CBC) algoritmy:

* 3des-cbc
* aes128-cbc
* aes192-cbc
* aes256-cbc
* blowfish-cbc
* cast128-cbc

Pre spojenia server-to-client sú povolené nasledovné Cipher Block Chaining (CBC) algoritmy:

* 3des-cbc
* aes128-cbc
* aes192-cbc
* aes256-cbc
* blowfish-cbc
* cast128-cbc

##### Riziko

Pre množstvo existujúcich zraniteľností, ktoré sú aplikovateľné len na algoritmy pracujúce v CBC móde, nie je CBC mód považovaný za bezpečný.

##### Odporúčanie

Odporúčame zablokovať na SSH serveri šifrovacie algoritmy v CBC móde a nahradiť ich algoritmami v módoch CTR alebo GCM.

### Služba SSL/TLS používa na zabezpečenie svojej komunikácie samo-podpísaný certifikát

| Riziko | ID zistenia | Zistenie |
| --- | --- | --- |
| Vysoké | 3.1.2 | Služba SSL/TLS používa na zabezpečenie svojej komunikácie samo-podpísaný certifikát |

##### Zistenie

Bolo zistené, že pre šifrovanú komunikáciu protokolmi SSL/TLS sú použité na strane servera samo-podpísané (self-signed) bezpečnostné certifikáty.

Zraniteľné systémy:

• cdf.Zákazník.sk, port 1 (TCP), port 2 (TCP), port 3 (TCP)

##### Riziko

Samo-podpísaný bezpečnostný certifikát neposkytuje záruku pôvodu tohto certifikátu. Tieto certifikáty sú ľahko falšovateľné a útočníkovi umožňujú realizovať útok typu „man-in-the-middle“, kedy sa útočník voči klientovi prezentuje ako server a voči serveru ako klient. Pri tomto type útoku má útočník plne pod kontrolou celú komunikáciu medzi klientom a serverom – môže ju ľubovoľne čítať aj modifikovať. Užívateľom sa síce zobrazí varovanie o použití takéhoto certifikátu, ale ak server bežne používa samo-podpísané certifikáty, užívatelia sú už zvyknutí takéto varovanie obísť a pokračovať v spojení.

Takýto útok je pomerne ťažko odhaliteľný. Legitímny užívateľ nemá možnosť si overiť autenticitu vygenerovaného certifikátu a tým pádom nemá možnosť zistiť, či sa pripája skutočne na server, na ktorý sa chcel pripojiť.

##### Odporúčanie

Odporúčame pre šifrovanú komunikáciu používať certifikáty vydané známou certifikačnou autoritou (internou alebo externou), ktorých pravosť je možné jednoznačne overiť a ktoré poskytujú dostatočnú záruku svojho pôvodu.

### Povolenie nešifrovaných spojení s POP3 serverom

| Riziko | ID zistenia | Zistenie |
| --- | --- | --- |
| Vysoké | 3.1.3 | Povolenie nešifrovaných spojení s POP3 serverom |

##### Zistenie

S POP3 serverom na testovanom systéme je možné komunikovať použitím nešifrovaného spojenia.

Zraniteľné systémy:

• kcx.Zákazník.sk, port 1 (TCP)

##### Riziko

Nezabezpečené (cleartext) sieťové protokoly prenášajú údaje v otvorenom tvare. To znamená, že ktokoľvek s prístupom k takejto sieťovej komunikácii má možnosť použitím voľne dostupných softvérových nástrojov sledovať, prípadne aj modifikovať prenášané údaje. To môže viesť k úniku citlivých údajov prípadne inému spôsobu narušenia bezpečnosti v dôsledku manipulácie prenášaných údajov.

##### Odporúčanie

Odporúčame zablokovať možnosť komunikácie s POP3 v otvorenej forme. Odporúčame POP3 komunikáciu chrániť TLS tunelom.

# Externý aplikačný penetračný test (mcx.Zákazník.sk)

## Aplikačná infraštruktúra

### Využívanie zraniteľnej JavaScript knižnice (AngularJS)

| Riziko | ID zistenia | Zistenie |
| --- | --- | --- |
| Nízke | 4.1.1 | Využívanie zraniteľnej JavaScript knižnice |

##### Zistenie

Bolo zistené, že aplikácia používa verziu 1.6.4 knižnice angularjs, ktorá má známe zraniteľnosti.

##### Riziko

Používanie JavaScript knižníc tretích strán môže priniesť celý rad zraniteľností založených na DOM, z ktorých, niektoré možno použiť na únos užívateľských účtov, ako je DOM-XSS.

Niektoré zraniteľnosti knižnice ohrozujú každú aplikáciu, ktorá importuje danú knižnicu, iné ovplyvňujú iba aplikácie, ktoré používajú určité funkcie knižnice.

##### Odporúčanie

Odporúčame aktualizovať uvedenú knižnicu na poslednú verziu (aktuálne je to 1.8.2).

## Autentifikácia

### Možnosť hádania prihlasovacích mien cez funkciu pre reset hesla

| Riziko | ID zistenia | Zistenie |
| --- | --- | --- |
| Stredné | 4.2.1 | Možnosť hádania prihlasovacích mien cez funkciu pre reset hesla |

##### Zistenie

Bolo zistené, funkcia pre reset hesla vracia rôznu odpoveď pre existujúce prihlasovacie meno a pre neexistujúce prihlasovacie meno.

Nasledujúci obrázok zobrazuje odpoveď aplikácie pri zadaní neexistujúceho prihlasovacieho mena:

**Obrázok bol odstranený.**

Nasledujúci obrázok zobrazuje odpoveď aplikácie pri zadaní existujúceho prihlasovacieho mena:

**Obrázok bol odstranený.**

##### Riziko

Táto vlastnosť aplikácie je zneužiteľná pre hádanie validných prihlasovacích mien. Identifikované prihlasovacie mená môžu byť následne použité pre hádanie hesiel cez prihlasovací formulár aplikácie.

##### Odporúčanie

Odporúčame upraviť odpovede aplikácie tak, aby vrátila rovnakú neutrálnu odpoveď aj pre existujúce aj pre neexistujúce prihlasovacie mená – napr. text “Ak bolo zadané prihlasovacie meno správne, na uvedenú e-mailovú adresu bola odoslaná linka pre reset hesla.”

## Správa relácie (Session Management)

### Identifikátor relácie prenášaný v URL

| Riziko | ID zistenia | Zistenie |
| --- | --- | --- |
| Stredné | 4.3.1 | Identifikátor relácie prenášaný v URL |

##### Zistenie

Zistili sme, že identifikátor klientskej relácie (AFHTOKEN) ktorý je nositeľom autentifikovanej identity používateľa je prenášaný v URL.

##### Riziko

Identifikátory relácie prenášané v URL môžu zostať uložené v „cache“ webového prehliadača používateľa kde môžu byť neskôr zneužité – hlavne v prípade zdieľaných počítačov. Ak aj daný počítač nie je zdieľaný uložené (a stále platné) identifikátory relácie môžu byť z „cache“ získané útočníkom, ktorý získal vzdialený prístup na daný počítač. Identifikátory relácie prenášané v URL sa tiež môžu dostať na iné systémy v „Referer“ hlavičkách pri prechode na inú web stránku. Navyše to umožňuje útočníkom jednoduchšie útoky cez nastrčené linky.

##### Odporúčanie

Odporúčame identifikátory relácie prenášať v tzv. HTTP cookie, alebo skrytých polí formulárov prenášaných použitím HTTP metódy POST.

## Validácia vstupov

### Možnosť vloženia dát vo formáte JSON na strane klienta

| Riziko | ID zistenia | Zistenie |
| --- | --- | --- |
| Vysoké | 4.4.1 | Možnosť vloženia dát vo formáte JSON na strane klienta |

##### Zistenie

Bolo zistené, že aplikácia posiela klientovi (webový prehliadač) rôzne konfiguračné nastavenia v dátových štruktúrach JSON (JavaScript Object Notation), ktoré sú následne na strane klienta spracovávané (parsované) funkciou JSON.parse()

Zdrojový prvok v kóde, ktorý obsahuje uvedenú dátovú štruktúru JSON má identifikátor “application-config”.

Do zdrojového prvku je vkladaná nasledujúca hodnota:

{"rootPath":"/", "apiPath":"/api/", "ws":"wss://portal.Zákazník.sk/ws"}

Uvedená hodnota obsahuje údaje, ktoré sú ovplyvniteľné užívateľom.

URL s výskytom uvedeného kódu na adrese portal.Zákazník.sk

* /login
* /password-reset

##### Riziko

Vkladanie (injekcia) JSON založená na DOM (Document Object Model) vzniká, keď skript vkladá užívateľom kontrolovateľné údaje do reťazca, ktorý je analyzovaný ako dátová štruktúra JSON a potom spracovaný aplikáciou. Útočník môže byť schopný použiť toto správanie na vytvorenie URL adresy, ktorá v prípade, že ju navštívi iný používateľ aplikácie, spôsobí spracovanie ľubovoľných údajov JSON. V závislosti od účelu, na ktorý sa tieto údaje používajú, môže byť možné narušiť logiku aplikácie alebo spôsobiť neúmyselné akcie v mene užívateľa.

Pri neautentifikovanom spôsobe testovania, ktorým prebiehali testy, bola k dispozícii len minimálna časť funkcií aplikácie, nebolo preto možné jednoznačne potvrdiť, či je uvedený kód zneužiteľný.

##### Odporúčanie

Najúčinnejším spôsobom, ako sa vyhnúť zraniteľnostiam vkladania JSON založených na DOM, je neanalyzovať ako JSON žiadny reťazec obsahujúci údaje pochádzajúce z nedôveryhodného zdroja. Ak požadovaná funkcionalita aplikácie spôsobuje, že tomuto správaniu sa nedá vyhnúť, v kóde na strane klienta je potrebné implementovať ochranu, aby sa zabránilo škodlivým údajom upravovať štruktúru JSON údajov nevhodným spôsobom. Môže to zahŕňať prísne overenie konkrétnych položiek, aby sa zabezpečilo, že neobsahujú žiadne znaky, ktoré by mohli zasahovať do štruktúry JSON pri jeho analyze.