



Universidade do Minho
Escola de Engenharia

Tecnologias de Segurança

Trabalho Prático 2

Grupo 4

Bruno Filipe de Sousa Dias PG47068
Guilherme da Silva Amorim Martins PG47225
Luís Enes Sousa A89597



4 de Maio de 2022

Conteúdo

1	Introdução	1
2	Análise e Prioritização dos riscos	2
2.1	CVSS - Common Vulnerability Scoring System	2
2.2	Matrix Risk Analysis	3
3	Modelação de Ameaças orientada aos Atacantes	4
3.1	Atacantes, Pontos Críticos e Pontos de Interesse	4
4	Portador	5
4.1	Android	6
4.1.1	CVE-2021-39708	6
4.1.2	CVE-2021-39694	7
4.1.3	CVE-2021-39698	8
4.2	IOS	9
4.2.1	CVE-2022-22667	9
4.2.2	CVE-2021-30996	10
4.2.3	CVE-2022-22633	11
4.3	Autenticação Móvel	12
5	Leitor/Verificador	13
5.1	Microsoft Windows	14
5.1.1	CVE-2020-1351	14
5.1.2	CVE-2020-0799	15
5.2	MacOS	16
5.2.1	CVE-2022-22657	16
5.2.2	CVE-2022-22638	17
6	Entidade Emissora	18
6.1	Backend principal: Django v3.0	19
6.1.1	CVE-2021-3281	19
6.1.2	CVE-2020-7471	20
6.1.3	CVE-2020-13254	21
6.2	Sistema operativo do servidor web: CentOS 7.8.2003	22
6.2.1	CVE-2020-5291	22
6.2.2	CVE-2020-10230	23
6.3	Servidor web: UWSGI	24
6.3.1	CVE-2018-7490	24
6.3.2	CVE-2018-6758	25
6.4	Base de dados principal: PostgreSQL 12.4	26
6.4.1	CVE-2021-32029	26
6.4.2	CVE-2020-25695	27
6.5	Backend de gestão: Flask 1.0	28
6.5.1	CVE-2022-24880	28
6.5.2	CVE-2018-16516	29
7	Relações entre Entidades	30
7.1	Portador - Entidade Emissora	30
7.1.1	Attack Man-In-The-Middle	30
7.1.2	IP spoofing	31
7.1.3	JSON Hijacking	31
7.1.4	Falta de Autenticação	32
7.2	Verificador/Leitor - Entidade Emissora	32

7.3	Portador - Verificador/Leitor	33
7.3.1	BLE - Bluetooth Low Energy	34
7.3.2	NFC - Near Field Communication	36
7.3.3	CBOR - Concise Binary Object Representation	37
7.4	Excesso de Informação	38
8	Recomendações de Segurança	39
8.1	Login e Código de acesso	39
8.2	SQL Injection Protection	39
8.3	Assinatura de Mensagens e Encriptação (Portador - Verificador) . . .	40
8.4	Sistema de Armazenamento	41
8.5	Proteção Binária (Rooting/Jailbreak)	41
9	Conclusão	42

Lista de Figuras

1	Matriz de Análise de Risco	3
2	Descrição CVE-2021-39708 na plataforma MITRE	6
3	Classificação CVE-2021-39708 plataforma NVD	6
4	Métricas CVE-2021-39708 plataforma NVD	6
5	Descrição CVE-2021-39694 na plataforma MITRE	7
6	Classificação CVE-2021-39694 plataforma NVD	7
7	Métricas CVE-2021-39694 plataforma NVD	7
8	Descrição CVE-2021-39698 na plataforma MITRE	8
9	Classificação CVE-2021-39698 plataforma NVD	8
10	Métricas CVE-2021-39698 plataforma NVD	8
11	Descrição CVE-2022-22667 na plataforma MITRE	9
12	Classificação CVE-2022-22667 plataforma NVD	9
13	Métricas CVE-2022-22667 plataforma NVD	9
14	Descrição CVE-2021-30996 na plataforma MITRE	10
15	Classificação CVE-2021-30996 plataforma NVD	10
16	Métricas CVE-2021-30996 plataforma NVD	10
17	Descrição CVE-2022-22633 na plataforma MITRE	11
18	Classificação CVE-2022-22633 plataforma NVD	11
19	Métricas CVE-2022-22633 plataforma NVD	11
20	Descrição CVE-2020-1351 na plataforma MITRE	14
21	Classificação CVE-2020-1351 plataforma NVD	14
22	Métricas CVE-2020-1351 plataforma NVD	14
23	Descrição CVE-2020-0799 na plataforma MITRE	15
24	Classificação CVE-2020-0799 plataforma NVD	15
25	Métricas CVE-2020-0799 plataforma NVD	15
26	Descrição CVE-2022-22657 na plataforma MITRE	16
27	Classificação CVE-2022-22657 plataforma NVD	16
28	Métricas CVE-2022-22657 plataforma NVD	16
29	Descrição CVE-2022-22638 na plataforma MITRE	17
30	Classificação CVE-2022-22638 plataforma NVD	17
31	Métricas CVE-2022-22638 plataforma NVD	17
32	Descrição CVE-2021-3281 na plataforma MITRE	19
33	Classificação CVE-2021-3281 plataforma NVD	19
34	Métricas CVE-2021-3281 plataforma NVD	19
35	Descrição CVE-2020-7471 na plataforma MITRE	20
36	Classificação CVE-2020-7471 plataforma NVD	20
37	Métricas CVE-2020-7471 plataforma NVD	20
38	Descrição CVE-2020-13254 na plataforma MITRE	21
39	Classificação CVE-2020-13254 plataforma NVD	21
40	Métricas CVE-2020-13254 plataforma NVD	21
41	Descrição CVE-2020-5291 na plataforma MITRE	22
42	Classificação CVE-2020-5291 plataforma NVD	22
43	Métricas CVE-2020-5291 plataforma NVD	22
44	Descrição CVE-2020-10230 na plataforma MITRE	23
45	Classificação CVE-2020-10230 plataforma NVD	23
46	Métricas CVE-2020-10230 plataforma NVD	23
47	Descrição CVE-2018-7490 na plataforma MITRE	24
48	Classificação CVE-2018-7490 plataforma NVD	24
49	Métricas CVE-2018-7490 plataforma NVD	24
50	Descrição CVE-2018-6758 na plataforma MITRE	25
51	Classificação CVE-2018-6758 plataforma NVD	25
52	Métricas CVE-2018-6758 plataforma NVD	25

53	Descrição CVE-2021-32029 na plataforma MITRE	26
54	Classificação CVE-2021-32029 plataforma NVD	26
55	Métricas CVE-2021-32029 plataforma NVD	26
56	Descrição CVE-2020-25695 na plataforma MITRE	27
57	Classificação CVE-2020-25695 plataforma NVD	27
58	Métricas CVE-2020-25695 plataforma NVD	27
59	Descrição CVE-2022-24880 na plataforma MITRE	28
60	Classificação CVE-2022-24880 plataforma NVD	28
61	Métricas CVE-2022-24880 plataforma NVD	28
62	Descrição CVE-2018-16516 na plataforma MITRE	29
63	Classificação CVE-2018-16516 plataforma NVD	29
64	Métricas CVE-2018-16516 plataforma NVD	29
65	Descrição CVE-2019-19196 na plataforma MITRE	34
66	Classificação CVE-2019-19196 plataforma NVD	34
67	Métricas CVE-2019-19196 plataforma NVD	34
68	Descrição CVE-2020-10061 na plataforma MITRE	35
69	Classificação CVE-2020-10061 plataforma NVD	35
70	Métricas CVE-2020-10061 plataforma NVD	35
71	Descrição CVE-2020-24753 na plataforma MITRE	37
72	Classificação CVE-2020-24753 plataforma NVD	37
73	Métricas CVE-2020-24753 plataforma NVD	37
74	Matriz de Análise de Risco	43

1 Introdução

No âmbito da unidade curricular de Tecnologias de Segurança, foi-nos proposto um trabalho prático onde agimos como membros da equipa de desenvolvimento de um serviço de identificação digital e móvel. Nesta equipa somos responsáveis pela componente de projeto relacionada com a segurança da informação e infraestrutura de suporte.

Faço ao que foi pedido, a equipa inicializa este projeto estruturando a maneira como deve ser feita a análise e prioritização dos riscos encontrados no Sistema. Seguidamente efetua uma modelação de ameaças (no nosso caso orientada aos Atacantes) e tenta perceber quais os Pontos de Interesse dos atacantes e os Pontos críticos do Sistema. Em seguida, efetua um estudo sobre as vulnerabilidades encontradas nas três entidades do Sistema (Portador, Verificador e Entidade Emissora), bem como nas comunicações e interações que estas possam ter ao longo de todo o processo. Finalmente serão ainda apresentadas algumas recomendações de segurança para o Sistema.

2 Análise e Prioritização dos riscos

Primeiramente, é preciso estabelecer parâmetros que permitam à equipa uma melhor organização face aos problemas de Segurança que vamos encontrar ao longo do caminho. Para isso, iremos medir os mesmos numa escala fixa, de modo a podermos preterir a resolução e mitigação de alguns riscos e vulnerabilidades em relação a outros, dependendo do seu nível de risco.

2.1 CVSS - Common Vulnerability Scoring System

Ao longo deste estudo iremos encontrar algumas vulnerabilidades passíveis de possuir um CVE na plataforma MITRE, e que por essa razão possuem uma descrição total da mesma e ainda um conjunto de métricas fornecidas pelo **CVSS (Common Vulnerability Scoring System)**. Esta framework é capaz de identificar e descrever características de uma vulnerabilidade, bem como da sua gravidade. Assim, as vulnerabilidades podem possuir os seguintes Ratings:

- **Critical** - Vulnerabilidades críticas do sistema. Score do CVSS encontra-se entre os valores 9.0 - 10.0. São vulnerabilidades que pedem a resolução imediata do seu problema, dado que colocam em grande risco o sistema.
- **High** - Vulnerabilidades de alto nível do sistema. Score do CVSS encontra-se entre os valores 7.0 - 8.9. São vulnerabilidades de colocam em risco o sistema numa escala alta, sem no entanto serem passíveis de ser classificadas como nível crítico, apesar da sua utilização poder trazer elevados danos.
- **Medium** - Vulnerabilidades de nível médio do sistema. Score do CVSS encontra-se entre os valores 4.0 - 6.9. São vulnerabilidades que colocam um risco médio no sistema. As vulnerabilidades que possuem este rating, são, normalmente, as mais comuns nos sistemas desenvolvidos e produzidos ao longo dos anos.
- **Low** - Vulnerabilidades de baixo nível do sistema. Score do CVSS encontra-se entre os valores 0.1 - 3.9. São vulnerabilidades encontradas que, apesar de não trazerem elevados problemas a um sistema, podem, quando devidamente exploradas, revelar informações a *attackers*, bem como ser aproveitadas para efetuar operações maliciosas.
- **Info** - Neste caso, e olhando para os parâmetros CVSS, não consideramos estes achados como vulnerabilidades em si. Score do CVSS é de 0.0. Não possuem qualquer perigo inerente ao sistema. No entanto, identificam que algumas informações do sistema e/ou máquina podem ser descobertas através da exploração de tais serviços. Identifica então, quais os serviços que são considerados como vulneráveis.

A resolução de vulnerabilidades deste tipo, será prioritizada da maneira como as mesmas foram apresentadas, sendo resolvidas as vulnerabilidades consoantes os seus Ratings pela seguinte ordem: **Critical - High - Medium - Low - Info**. Iremos ainda reparar que no estudo feito não apresentamos nenhuma vulnerabilidade com Rating Score de 0.0 (Info), dado não acharmos relevante para esta fase de estudo do Sistema.

2.2 Matrix Risk Analysis

Vimos já que vamos estudar vulnerabilidades passíveis de possuir um CVE na plataforma MITRE. No entanto, iremos também estudar vulnerabilidades que não estão presentes nesta plataforma, dado maioritariamente a serem muito mais abrangentes. Para identificarmos, classificarmos e priorizarmos as diferentes vulnerabilidades iremos utilizar como recurso a tabela apresentada abaixo na Figura 1 (também disponível no Anexo A - Figura 74).

		SEVERITY				RISK RATING KEY
		Acceptable	Tolerable	Undesirable	Intolerable	
LIKELIHOOD	Improbable Risk is Unlikely to Occur	1 LOW	4 MEDIUM	7 MEDIUM	10 HIGH	LOW Ok to Proceed
	Possible Risk is Likely to Occur	2 LOW	5 MEDIUM	8 HIGH	11 EXTREME	MEDIUM Take Mitigation Effort
	Probable Risk Will Occur	3 MEDIUM	6 HIGH	9 HIGH	12 EXTREME	HIGH SEEK SUPPORT
						EXTREME PLACE EVENT ON HOLD

Figura 1: Matriz de Análise de Risco

Nesta tabela teremos então presentes duas características diferentes, **Likelihood** e **Severity**. A Likelihood representa a probabilidade de uma certa vulnerabilidade ser explorada, A Severity representa a gravidade que a exploração de uma dada vulnerabilidade pode ter, bem como os problemas que pode causar a uma empresa.

Para calcular o risco teremos de fazer a simples conta de: **Risk = Likelihood x Severity**. Esta conta irá fornecer desta forma quantitativamente, e por isso, qualitativamente o nível risco com que estamos a lidar. Também aqui teremos 4 níveis de risco: **Low, Medium, High, Extreme**. Todos eles seguem o mesmo padrão do que os Ratings de CVSS apresentados na secção anterior (Extreme desta matriz, equivala Critical do CVSS).

Mais uma vez a resolução de vulnerabilidades será priorizada consoantes os diferentes níveis de risco das mesmas, seguindo a seguinte ordem: **Extreme - High - Medium - Low**.

Sabemos ainda que quanto à **Likelihood** possuímos 3 níveis possíveis: **Probable** – Muito provável de acontecer; **Possible** – Alguma chance de acontecer; **Improbable** – Pequena chance de acontecer.

Quanto à **Severity** possuímos 4 níveis diferentes: **Acceptable** – pouco ou nenhum efeito numa empresa; **Tolerable** – os efeitos são sentidos, mas não afetam seriamente uma empresa; **Undesirable** – causa grande interrupção numa empresa; **Intolerable** – uma empresa pode não conseguir recuperar.

3 Modelação de Ameaças orientada aos Atacantes

Neste momento iremos tentar perceber e estudar a maneira como a nossa aplicação e sistema podem ser vulneráveis. A melhor maneira para começar este processo será perceber de onde poderão vir os ataques ao nosso sistema e quem os poderá efetuar. A partir daí, poderemos começar a estudar as fraquezas e vulnerabilidades do sistema em construção.

3.1 Atacantes, Pontos Críticos e Pontos de Interesse

Começamos então por analisar quem são os atacantes comuns que poderemos esperar no nosso sistema em desenvolvimento. Para isso, teremos de tentar perceber principalmente do que tratará o nosso sistema e tentar perceber o porquê de os *attackers* quererem efetuar ataques ao mesmo.

Sabemos que estamos a trabalhar num Serviço de Identificação Digital e Móvel. Posto isto, podemos concluir que o nosso Sistema irá lidar com muita informação sensível e privada de uma vastidão de indivíduos diferentes. Os documentos, dados e informações pessoais de inúmeras pessoas irá "estar em jogo". Podemos concluir com toda a certeza que ataques maliciosos efetuados com sucesso a este Sistema terão repercussões de um nível bastante elevado. Poderão surgir roubos de identidade ou venda e contrabando de informação, por exemplo.

Considerando as conclusões obtidas quanto ao tipo de informação a tratar neste Sistema, bem como das adversidades que podem existir se um ataque for bem sucedido, chegamos agora ao momento de perceber, quem poderia querer efetuar tais ataques. Sendo assim, teremos atacantes do tipo:

- **Black Hat Hackers** - Especialistas no mundo da Informática, que possuem uma intenção errada;
- **Empresas Concorrentes** - Empresas que competem para ter uma melhor aplicação do que a da nossa Empresa, dado que as duas seguem os mesmos moldes e possuem as mesmas funcionalidades;

Sendo assim, teremos duas razões principais para ataques. Uma delas será o roubo de identidades, ou de informações sensíveis sobre indivíduos - podemos imaginar que o atacante roube informações sobre um indivíduo com uma grande força política, muito famoso ou até incrivelmente rico, de forma a apoderar-se da sua identidade, ou de vender tais informações obtidas. Uma segunda razão será, poder estudar a aplicação para roubar ideias da mesmas, ou até mesmo, tentar sabotar o Sistema e aplicação, de modo à empresa não conseguir recuperar e a Empresa Concorrente começar a ter muito mais sucesso.

Sendo assim, podemos perceber que os Pontos de Interesse dos atacantes neste Sistema são todos os que lhe podem revelar informações importantes sobre os indivíduos. Desta forma, os pontos de interesse serão as três entidades: Portador, Verificador e Entidade Emissora. Para além das entidades, também as ligações e comunicações efetuadas entre elas serão pontos de interesse a um atacante.

Como Pontos Críticos deste Sistema, poderemos considerar também todas as três Entidades (Portador, Verificador e Entidade Emissora) que de si fazem parte, bem como as comunicações estabelecidas entre as mesmas.

Sendo assim, estamos então preparados para começar a estudar as vulnerabilidades presentes tanto nas Entidades do Sistema (Portador, Verificador e Entidade Emissora), bem como nas comunicações presentes nas comunicações estabelecidas entre estas entidades.

4 Portador

Iremos nesta secção estudar o portador, ou seja, o cidadão com acesso a um *smartphone* que irá armazenar a mID.

Sabemos antecipadamente que este portador irá executar uma aplicação mID, sendo esta uma aplicação móvel para sistemas Android e/ou IOS. Várias informações adicionais foram reveladas, no entanto, estas remetem mais para a sua comunicação com as outras entidades, e por isso apenas serão discutidas brevemente.

Desta forma iremos inicialmente estudar vulnerabilidades que possam existir em sistemas Android e IOS (sistemas onde aplicação irá correr) que possam pôr em causa a integridade do Sistema.

Nos casos em que não seja possível analisar uma vulnerabilidade através do **CVSS (Common Vulnerability Scoring System)**, iremos fazer a nossa própria análise (não havendo desta forma uma precisão tão grande).

4.1 Android

Iremos inicialmente estudar os sistemas Android que podem ser utilizados para a operação desta aplicação, identificando as suas principais vulnerabilidades.

4.1.1 CVE-2021-39708

A primeira vulnerabilidade encontrada foi publicada no **MITRE** no dia 23 de Agosto de 2021, possuindo como CVE-ID, **CVE-2021-39708**. A vulnerabilidade encontrada indica que em *gatt_process_notification* do arquivo *gatt.cl.cc* existe uma possível gravação de conteúdo fora dos limites devido a uma verificação de limites incorreta. Isto pode levar ao escalonamento remoto de privilégios sem a necessidade de privilégios de execução adicionais.

CVE-ID	
CVE-2021-39708	Learn more at National Vulnerability Database (NVD)
• CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information	
Description	
In gatt_process_notification of gatt.cl.cc, there is a possible out of bounds write due to an incorrect bounds check. This could lead to remote escalation of privilege with no additional execution privileges needed. User interaction is not needed for exploitation.Product: AndroidVersions: Android-12Android ID: A-206128341	

Figura 2: Descrição CVE-2021-39708 na plataforma MITRE

Analisando agora as métricas, podemos perceber que esta vulnerabilidade possui um *Base Score* de 9.8, sendo considerada como uma vulnerabilidade de nível crítico. Sabemos ainda que o seu impacto é de 5.9 e o *score* da sua *exploitability* é de 3.9. Quanto ao ataque, podemos perceber alguns aspetos pela informação recolhida:

- Tem de ser efetuado na Rede;
- É um ataque de complexidade baixa;
- Não é necessário qualquer tipo de privilégio ou interação com o Utilizador;
- Confidencialidade, Integridade e Disponibilidade afetadas com um grau de impacto alto.

CVE-2021-39708 Detail


Current Description

In gatt_process_notification of gatt.cl.cc, there is a possible out of bounds write due to an incorrect bounds check. This could lead to remote escalation of privilege with no additional execution privileges needed. User interaction is not needed for exploitation.Product: AndroidVersions: Android-12Android ID: A-206128341

[+New Analysis Description](#)

Severity CVSS Version 3.x CVSS Version 2.0

CVSS 3.x Severity and Metrics:

 NIST: NVD

Base Score: **9.8 CRITICAL**

Vector: CVSS:3.1/(AV:N)/(AC:L)/(PR:N)/(UI:N)/(S:U)/(C:H)/(I:H)/(A:H)

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

CVSS v3.1 Severity and Metrics:

Base Score: 9.8 CRITICAL

Vector: AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

Impact Score: 5.9

Exploitability Score: 3.9

Attack Vector (AV): Network

Attack Complexity (AC): Low

Privileges Required (PR): None

User Interaction (UI): None

Scope (S): Unchanged

Confidentiality (C): High

Integrity (I): High

Availability (A): High

Figura 3: Classificação CVE-2021-39708 plataforma NVD

Figura 4: Métricas CVE-2021-39708 plataforma NVD

4.1.2 CVE-2021-39694

A segunda vulnerabilidade encontrada foi publicada no **MITRE** no dia 23 de Agosto de 2021, possuindo como CVE-ID, **CVE-2021-39694**. A vulnerabilidade encontrada indica que, analisando o ficheiro RoleParsers.Java, podemos perceber que existe uma maneira possível de aplicações padrão obterem permissões explicitamente negadas pelo Utilizador devido a um *bypass* de permissões. Isto pode levar ao escalonamento remoto de privilégios sem a necessidade de privilégios de execução adicionais.

CVE-ID	
CVE-2021-39694	Learn more at National Vulnerability Database (NVD)
• CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information	
Description	
In parse of RoleParser.java, there is a possible way for default apps to get permissions explicitly denied by the user due to a permissions bypass. This could lead to local escalation of privilege with no additional execution privileges needed. User interaction is not needed for exploitation.Product: AndroidVersions: Android-12Android ID: A-202312327	

Figura 5: Descrição CVE-2021-39694 na plataforma MITRE

Analisando agora as métricas, podemos perceber que esta vulnerabilidade possui um *Base Score* de 7.8, sendo considerada como uma vulnerabilidade de nível alto. Sabemos ainda que o seu impacto é de 5.9 e o *score* da sua *exploitability* é de 1.8. Quanto ao ataque, podemos perceber alguns aspetos pela informação recolhida:

- Tem de ser efetuado localmente;
- É um ataque de complexidade baixa;
- É necessário um nível de privilégio baixo;
- Não é necessária qualquer interação com o Utilizador;
- Confidencialidade, Integridade e Disponibilidade afetadas com um grau de impacto alto.

CVE-2021-39694 Detail


Current Description

In parse of RoleParser.java, there is a possible way for default apps to get permissions explicitly denied by the user due to a permissions bypass. This could lead to local escalation of privilege with no additional execution privileges needed. User interaction is not needed for exploitation.Product: AndroidVersions: Android-12Android ID: A-202312327

[View Analysis Description](#)

Severity CVSS Version 3.x CVSS Version 2.0

CVSS 3.x Severity and Metrics:

 NIST: NVD

Base Score: **7.8 HIGH**

Vector: CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

CVSS v3.1 Severity and Metrics:

Base Score: 7.8 HIGH

Vector: AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H

Impact Score: 5.9

Exploitability Score: 1.8

Attack Vector (AV): Local

Attack Complexity (AC): Low

Privileges Required (PR): Low

User Interaction (UI): None

Scope (S): Unchanged

Confidentiality (C): High

Integrity (I): High

Availability (A): High

Figura 6: Classificação CVE-2021-39694 plataforma NVD

Figura 7: Métricas CVE-2021-39694 plataforma NVD

4.1.3 CVE-2021-39698

A terceira vulnerabilidade encontrada foi publicada no **MITRE** no dia 23 de Agosto de 2021, possuindo como CVE-ID, **CVE-2021-39698**. A vulnerabilidade encontrada indica que em *aio_poll_complete_work* do ficheiro *aio.c*, existe uma possível corrupção de memória devido a um uso depois de ser dado *free*. Isto pode levar ao escalonamento remoto de privilégios sem a necessidade de privilégios de execução adicionais.

CVE-ID	
CVE-2021-39698	Learn more at National Vulnerability Database (NVD)
• CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information	
Description	
In aio_poll_complete_work of aio.c, there is a possible memory corruption due to a use after free. This could lead to local escalation of privilege with no additional execution privileges needed. User interaction is not needed for exploitation.Product: AndroidVersions: Android kernelAndroid ID: A-185125206References: Upstream kernel	

Figura 8: Descrição CVE-2021-39698 na plataforma MITRE

Analisando agora as métricas, podemos perceber que esta vulnerabilidade possui um *Base Score* de 7.8, sendo considerada como uma vulnerabilidade de nível alto. Sabemos ainda que o seu impacto é de 5.9 e o *score* da sua *exploitability* é de 1.8. Quanto ao ataque, podemos perceber alguns aspetos pela informação recolhida:

- Tem de ser efetuado localmente;
- É um ataque de complexidade baixa;
- É necessário um nível de privilégio baixo;
- Não é necessária qualquer interação com o Utilizador;
- Confidencialidade, Integridade e Disponibilidade afetadas com um grau de impacto alto.

CVE-2021-39698 Detail


Current Description

In aio_poll_complete_work of aio.c, there is a possible memory corruption due to a use after free. This could lead to local escalation of privilege with no additional execution privileges needed. User interaction is not needed for exploitation.Product: AndroidVersions: Android kernelAndroid ID: A-185125206References: Upstream kernel

[View Analysis Description](#)

Severity CVSS Version 3.x CVSS Version 2.0

CVSS 3.x Severity and Metrics:

 NIST: NVD Base Score: **7.8 HIGH** Vector: CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

CVSS v3.1 Severity and Metrics:

Base Score: 7.8 HIGH
Vector: AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H
Impact Score: 5.9
Exploitability Score: 1.8

Attack Vector (AV): Local
Attack Complexity (AC): Low
Privileges Required (PR): Low
User Interaction (UI): None
Scope (S): Unchanged
Confidentiality (C): High
Integrity (I): High
Availability (A): High

Figura 9: Classificação CVE-2021-39698 plataforma NVD

Figura 10: Métricas CVE-2021-39698 plataforma NVD

4.2 IOS

Iremos agora estudar os sistemas IOS que podem ser utilizados para a operação desta aplicação, identificando as suas principais vulnerabilidades.

4.2.1 CVE-2022-22667

A primeira vulnerabilidade encontrada foi publicada no **MITRE** no dia 5 de Janeiro de 2022, possuindo como CVE-ID, **CVE-2022-22667**. A vulnerabilidade encontrada indica que existe uma possível corrupção e/ou problema de memória devido a um uso depois de ser dado *free*. Uma aplicação poderia desta forma executar código arbitrário com privilégios *kernel*.

CVE-ID
CVE-2022-22667 Learn more at National Vulnerability Database (NVD)
• CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information
Description
A use after free issue was addressed with improved memory management. This issue is fixed in iOS 15.4 and iPadOS 15.4. An application may be able to execute arbitrary code with kernel privileges.

Figura 11: Descrição CVE-2022-22667 na plataforma MITRE

Analisando agora as métricas, podemos perceber que esta vulnerabilidade possui um *Base Score* de 7.8, sendo considerada como uma vulnerabilidade de nível alto. Sabemos ainda que o seu impacto é de 5.9 e o *score* da sua *exploitability* é de 1.8. Quanto ao ataque, podemos perceber alguns aspetos pela informação recolhida:

- Tem de ser efetuado localmente;
- É um ataque de complexidade baixa;
- Não é necessário qualquer um nível de privilégio;
- É necessária interação com o Utilizador;
- Confidencialidade, Integridade e Disponibilidade afetadas com um grau de impacto alto.

CVE-2022-22667 Detail


Current Description

A use after free issue was addressed with improved memory management. This issue is fixed in iOS 15.4 and iPadOS 15.4. An application may be able to execute arbitrary code with kernel privileges.

[View Analysis Description](#)

Severity CVSS Version 3.x CVSS Version 2.0

CVSS 3.x Severity and Metrics:

 **NIST: NVD** **Base Score: 7.8 HIGH** **Vector: CVSS:3.1/AV:L/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H**

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

CVSS v3.1 Severity and Metrics:

Base Score: 7.8 HIGH

Vector: AV:L/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H

Impact Score: 5.9

Exploitability Score: 1.8

Attack Vector (AV): Local

Attack Complexity (AC): Low

Privileges Required (PR): None

User Interaction (UI): Required

Scope (S): Unchanged

Confidentiality (C): High

Integrity (I): High

Availability (A): High

Figura 12: Classificação CVE-2022-22667 plataforma NVD

Figura 13: Métricas CVE-2022-22667 plataforma NVD

4.2.2 CVE-2021-30996

A segunda vulnerabilidade encontrada foi publicada no **MITRE** no dia 13 de Abril de 2021, possuindo como CVE-ID, **CVE-2021-30996**. A vulnerabilidade encontrada indica que existe uma *race condition* e com a qual um aplicativo malicioso pode executar código arbitrário com privilégios de *kernel*.

CVE-ID
CVE-2021-30996 Learn more at National Vulnerability Database (NVD) • CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information
Description
A race condition was addressed with improved state handling. This issue is fixed in macOS Monterey 12.1, iOS 15.2 and iPadOS 15.2. A malicious application may be able to execute arbitrary code with kernel privileges.

Figura 14: Descrição CVE-2021-30996 na plataforma MITRE

Analisando agora as métricas, podemos perceber que esta vulnerabilidade possui um *Base Score* de 7.0, sendo considerada como uma vulnerabilidade de nível alto. Sabemos ainda que o seu impacto é de 5.9 e o *score* da sua *exploitability* é de 1.0. Quanto ao ataque, podemos perceber alguns aspetos pela informação recolhida:

- Tem de ser efetuado localmente;
- É um ataque de complexidade alta;
- Não é necessário qualquer um nível de privilégio;
- É necessária interação com o Utilizador;
- Confidencialidade, Integridade e Disponibilidade afetadas com um grau de impacto alto.


CVE-2021-30996 Detail

Current Description
A race condition was addressed with improved state handling. This issue is fixed in macOS Monterey 12.1, iOS 15.2 and iPadOS 15.2. A malicious application may be able to execute arbitrary code with kernel privileges.

[View Analysis Description](#)

Severity CVSS Version 3.x CVSS Version 2.0

CVSS 3.x Severity and Metrics:

 NIST: NVD Base Score: **7.0 HIGH** Vector: CVSS:3.1/AV:L/AC:H/PR:N/UI:R/S:U/C:H/I:H/A:H

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

Figura 15: Classificação CVE-2021-30996 plataforma NVD

CVSS v3.1 Severity and Metrics:

Base Score: 7.0 HIGH

Vector: AV:L/AC:H/PR:N/UI:R/S:U/C:H/I:H/A:H

Impact Score: 5.9

Exploitability Score: 1.0

Attack Vector (AV): Local

Attack Complexity (AC): High

Privileges Required (PR): None

User Interaction (UI): Required

Scope (S): Unchanged

Confidentiality (C): High

Integrity (I): High

Availability (A): High

Figura 16: Métricas CVE-2021-30996 plataforma NVD

4.2.3 CVE-2022-22633

A terceira vulnerabilidade encontrada foi publicada no **MITRE** no dia 5 de Janeiro de 2022, possuindo como CVE-ID, **CVE-2022-22633**. A vulnerabilidade encontrada indica que existe um problema de corrupção de memória. Ao abrir um arquivo PDF com código malicioso pode haver um encerramento inesperado de aplicações e ainda a possível execução de código arbitrário.

CVE-ID	
CVE-2022-22633	Learn more at National Vulnerability Database (NVD)
• CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information	
Description	
A memory corruption issue was addressed with improved state management. This issue is fixed in watchOS 8.5, iOS 15.4 and iPadOS 15.4, macOS Big Sur 11.6.5, macOS Monterey 12.3. Opening a maliciously crafted PDF file may lead to an unexpected application termination or arbitrary code execution.	

Figura 17: Descrição CVE-2022-22633 na plataforma MITRE

Analisando agora as métricas, podemos perceber que esta vulnerabilidade possui um *Base Score* de 7.8, sendo considerada como uma vulnerabilidade de nível alto. Sabemos ainda que o seu impacto é de 5.9 e o *score* da sua *exploitability* é de 1.8. Quanto ao ataque, podemos perceber alguns aspetos pela informação recolhida:

- Tem de ser efetuado localmente;
- É um ataque de complexidade baixa;
- Não é necessário qualquer um nível de privilégio;
- É necessária interação com o Utilizador;
- Confidencialidade, Integridade e Disponibilidade afetadas com um grau de impacto alto.

CVE-2022-22633 Detail


Current Description

A memory corruption issue was addressed with improved state management. This issue is fixed in watchOS 8.5, iOS 15.4 and iPadOS 15.4, macOS Big Sur 11.6.5, macOS Monterey 12.3. Opening a maliciously crafted PDF file may lead to an unexpected application termination or arbitrary code execution.

[View Analysis Description](#)

Severity CVSS Version 3.x CVSS Version 2.0

CVSS 3.x Severity and Metrics:

 NIST: NVD

Base Score: **7.8 High**

Vector: CVSS:3.1/(WPL)/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

CVSS v3.1 Severity and Metrics:

Base Score: 7.8 HIGH

Vector: AV:L/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H

Impact Score: 5.9

Exploitability Score: 1.8

Attack Vector (AV): Local

Attack Complexity (AC): Low

Privileges Required (PR): None

User Interaction (UI): Required

Scope (S): Unchanged

Confidentiality (C): High

Integrity (I): High

Availability (A): High

Figura 18: Classificação CVE-2022-22633 plataforma NVD

Figura 19: Métricas CVE-2022-22633 plataforma NVD

4.3 Autenticação Móvel

Para além das vulnerabilidades encontradas acima, pode ainda existir uma grande vulnerabilidade no dispositivo no dispositivo Portador. Este não está diretamente ligado ao Sistema em estudo, no entanto poderá ter danos colaterais.

Hoje em dia a Segurança Digital tem vindo a aumentar cada vez mais. No entanto, nem todos nós somos bons praticantes das técnicas que nos permitem a nossa defesa no mundo tecnológico no dia a dia. Muitas vezes, existem cidadãos que não possuem qualquer tipo de password ou autenticação no seu *smartphone*, sendo apenas necessário um *swipe* para desbloquear o mesmo e ter o seu acesso total (embora que pouco provável). Para além disso, é costume ser aplicada uma password e/ou uma autenticação bastante simples para desbloquear o mesmo (por exemplo, um padrão que faz a letra L ou um código que é 1234). Também estas passwords precárias tornam o nosso sistema mais fácil de penetrar e atacar, sendo apenas necessário um simples *brute-force*.

Nestes casos, ao perder o nosso *smartphone*, ou até mesmo, se este for roubado, quem tomar posse do mesmo terá o controlo total dele. Dessa forma, o sistema em estudo torna-se vulnerável, havendo uma brecha, ainda que não tecnológica, para efetuar um ataque e obter informação que não desejamos.

Analisando a matriz de análise de risco do Anexo A (Figura 74), podemos determinar a "*likelihood*" deste ataque como **Improbable** e a "*severity*" como **Undesirable**, sendo desta forma um ataque com um risco de impacto médio, onde iremos precisar de procurar ajuda para resolver o mesmo.

A maneira mais forte de prevenir ataques deste tipo, é possuir uma password no *smartphone* com um elevado grau de qualidade e robustez.

5 Leitor/Verificador

Iremos nesta secção estudar o Leitor/Verificador, ou seja, a entidade com acesso a um *smartphone*, ou qualquer dispositivo que suporte os protocolos de comunicação e de operações de todo o sistema, que irá ser o responsável por verificar a autenticidade e veracidade de um documento mID do Portador.

Sabemos antecipadamente que este Leitor/Verificador irá executar uma aplicação leitora, sendo esta uma aplicação móvel para sistemas Android e/ou IOS ou qualquer dispositivo que suporte os protocolos de comunicação e de operações de todo o sistema. Várias informações adicionais foram reveladas, no entanto, estas remetem mais para a sua comunicação com as outras entidades, e por isso apenas serão discutidas brevemente.

Tal como o Portador, o Leitor/Verificador partilha as suas vulnerabilidades a nível de **Android**, **IOS** e ainda a nível de **autenticação móvel**. Desta forma iremos apenas estudar nesta entidade os sistemas adicionais de **Microsoft Windows** e **MacOS**, que possam pôr em causa a integridade do Sistema. É importante perceber que também os sistemas de Microsoft Windows e MacOS partilham da vulnerabilidade da autenticação móvel apresentada anteriormente.

Nos casos em que não seja possível analisar uma vulnerabilidade através do **CVSS (Common Vulnerability Scoring System)**, iremos fazer a nossa própria análise (não havendo desta forma uma precisão tão grande).

5.1 Microsoft Windows

Iremos inicialmente estudar o sistema operativo **Microsoft Windows**, identificando e estudando as suas principais vulnerabilidades.

5.1.1 CVE-2020-1351

A primeira vulnerabilidade encontrada foi publicada no **MITRE** no dia 4 de Novembro de 2019, possuindo como CVE-ID, **CVE-2020-1351**. A vulnerabilidade encontrada indica que existe uma divulgação de informações quando a componente do Windows Graphics manipula incorretamente objetos na memória, tornando assim esta informação vulnerável.

CVE-ID
CVE-2020-1351 Learn more at National Vulnerability Database (NVD)
CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information
Description
An information disclosure vulnerability exists when the Windows Graphics component improperly handles objects in memory, aka 'Microsoft Graphics Component Information Disclosure Vulnerability'.

Figura 20: Descrição CVE-2020-1351 na plataforma MITRE

Analisando agora as métricas, podemos perceber que esta vulnerabilidade possui um *Base Score* de 5.5, sendo considerada como uma vulnerabilidade de nível médio. Sabemos ainda que o seu impacto é de 3.6 e o *score* da sua *exploitability* é de 1.8. Quanto ao ataque, podemos perceber alguns aspetos pela informação recolhida:

- Tem de ser efetuado localmente;
- É um ataque de complexidade baixa;
- É necessário um nível baixo de privilégio;
- Não é necessária interação com o Utilizador;
- Confidencialidade afetada com um grau de impacto alto;
- Integridade e Disponibilidade não afetadas.

CVE-2020-1351 Detail


Current Description

An information disclosure vulnerability exists when the Windows Graphics component improperly handles objects in memory, aka 'Microsoft Graphics Component Information Disclosure Vulnerability'.

[View Analysis Description](#)

Severity CVSS Version 3.x CVSS Version 2.0

CVSS 3.x Severity and Metrics:

 **NIST: NVD** **Base Score:** 5.5 MEDIUM **Vector:** CVSS:3.1/(AV:L)/(AC:L)/(PR:L)/(UI:N)/(S:U)/(C:H)/(I:N)/(A:N)

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

CVSS v3.1 Severity and Metrics:

Base Score: 5.5 MEDIUM
Vector: AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N
Impact Score: 3.6
Exploitability Score: 1.8

Attack Vector (AV): Local
Attack Complexity (AC): Low
Privileges Required (PR): Low
User Interaction (UI): None
Scope (S): Unchanged
Confidentiality (C): High
Integrity (I): None
Availability (A): None

Figura 21: Classificação CVE-2020-1351 plataforma NVD

Figura 22: Métricas CVE-2020-1351 plataforma NVD

5.1.2 CVE-2020-0799

A segunda vulnerabilidade encontrada foi publicada no **MITRE** no dia 4 de Novembro de 2019, possuindo como CVE-ID, **CVE-2020-0799**. A vulnerabilidade encontrada indica que existe uma elevação de privilégio no Microsoft Windows quando o kernel não consegue lidar corretamente com a análise de certos *links* simbólicos.

CVE-ID
CVE-2020-0799 Learn more at National Vulnerability Database (NVD) • CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information
Description
An elevation of privilege vulnerability exists in Microsoft Windows when the Windows kernel fails to properly handle parsing of certain symbolic links, aka 'Windows Kernel Elevation of Privilege Vulnerability'.

Figura 23: Descrição CVE-2020-0799 na plataforma MITRE

Analisando agora as métricas, podemos perceber que esta vulnerabilidade possui um *Base Score* de 7.8, sendo considerada como uma vulnerabilidade de nível alto. Sabemos ainda que o seu impacto é de 5.9 e o *score* da sua *exploitability* é de 1.8. Quanto ao ataque, podemos perceber alguns aspetos pela informação recolhida:

- Tem de ser efetuado localmente;
- É um ataque de complexidade baixa;
- É necessário um nível baixo de privilégio;
- Não é necessária interação com o Utilizador;
- Confidencialidade, Integridade e Disponibilidade afetadas com um grau de impacto alto.

CVE-2020-0799 Detail

Current Description


An elevation of privilege vulnerability exists in Microsoft Windows when the Windows kernel fails to properly handle parsing of certain symbolic links, aka 'Windows Kernel Elevation of Privilege Vulnerability'.

[View Analysis Description](#)

Severity

CVSS Version 3.x CVSS Version 2.0

CVSS 3.x Severity and Metrics:

 NIST: NVD

Base Score: **7.8 HIGH**

Vector: CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

CVSS v3.1 Severity and Metrics:

Base Score: 7.8 HIGH

Vector: AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H

Impact Score: 5.9

Exploitability Score: 1.8

Attack Vector (AV): Local

Attack Complexity (AC): Low

Privileges Required (PR): Low

User Interaction (UI): None

Scope (S): Unchanged

Confidentiality (C): High

Integrity (I): High

Availability (A): High

Figura 24: Classificação CVE-2020-0799 plataforma NVD

Figura 25: Métricas CVE-2020-0799 plataforma NVD

5.2 MacOS

Iremos inicialmente estudar o sistema operativo **MacOS**, identificando e estudando as suas principais vulnerabilidades.

5.2.1 CVE-2022-22657

A primeira vulnerabilidade encontrada foi publicada no **MITRE** no dia 5 de Janeiro de 2022, possuindo como CVE-ID, **CVE-2022-22657**. A vulnerabilidade encontrada indica que existe um problema de inicialização de memória, onde quando um arquivo malicioso é aberto, pode levar ao encerramento repentino de aplicações, bem como à execução de código arbitrário.

CVE-ID
CVE-2022-22657 Learn more at National Vulnerability Database (NVD)
CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information
Description
A memory initialization issue was addressed with improved memory handling. This issue is fixed in Logic Pro 10.7.3, GarageBand 10.4.6, macOS Monterey 12.3. Opening a maliciously crafted file may lead to unexpected application termination or arbitrary code execution.

Figura 26: Descrição CVE-2022-22657 na plataforma MITRE

Analisando agora as métricas, podemos perceber que esta vulnerabilidade possui um *Base Score* de 7.8, sendo considerada como uma vulnerabilidade de nível alto. Sabemos ainda que o seu impacto é de 5.9 e o *score* da sua *exploitability* é de 1.8. Quanto ao ataque, podemos perceber alguns aspetos pela informação recolhida:

- Tem de ser efetuado localmente;
- É um ataque de complexidade baixa;
- Não é necessário qualquer nível de privilégio;
- É necessária interação com o Utilizador;
- Confidencialidade, Integridade e Disponibilidade afetadas com um grau de impacto alto.

CVE-2022-22657 Detail

Current Description

A memory initialization issue was addressed with improved memory handling. This issue is fixed in Logic Pro 10.7.3, GarageBand 10.4.6, macOS Monterey 12.3. Opening a maliciously crafted file may lead to unexpected application termination or arbitrary code execution.


[View Analysis Description](#)

Severity

CVSS Version 3.x

CVSS Version 2.0

CVSS 3.x Severity and Metrics:

 NIST: NVD

Base Score: **7.8 HIGH**

Vector: CVSS:3.1/AV:L/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.
Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

CVSS v3.1 Severity and Metrics:

Base Score: 7.8 HIGH

Vector: AV:L/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H

Impact Score: 5.9

Exploitability Score: 1.8

Attack Vector (AV): Local

Attack Complexity (AC): Low

Privileges Required (PR): None

User Interaction (UI): Required

Scope (S): Unchanged

Confidentiality (C): High

Integrity (I): High

Availability (A): High

Figura 27: Classificação CVE-2022-22657 plataforma NVD

Figura 28: Métricas CVE-2022-22657 plataforma NVD

5.2.2 CVE-2022-22638

A segunda vulnerabilidade encontrada foi publicada no **MITRE** no dia 5 de Janeiro de 2022, possuindo como CVE-ID, **CVE-2022-22638**. A vulnerabilidade encontrada indica que existe um erro com um *null pointer*, a partir do qual um atacante poderia realizar um ataque do tipo *denial-of-service*.

CVE-ID
CVE-2022-22638 Learn more at National Vulnerability Database (NVD)
• CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information
Description
A null pointer dereference was addressed with improved validation. This issue is fixed in tvOS 15.4, iOS 15.4 and iPadOS 15.4, macOS Big Sur 11.6.5, Security Update 2022-003 Catalina, watchOS 8.5, macOS Monterey 12.3. An attacker in a privileged position may be able to perform a denial of service attack.

Figura 29: Descrição CVE-2022-22638 na plataforma MITRE

Analisando agora as métricas, podemos perceber que esta vulnerabilidade possui um *Base Score* de 6.5, sendo considerada como uma vulnerabilidade de nível médio. Sabemos ainda que o seu impacto é de 3.6 e o *score* da sua *exploitability* é de 2.8. Quanto ao ataque, podemos perceber alguns aspetos pela informação recolhida:

- Tem de ser efetuado na Rede;
- É um ataque de complexidade baixa;
- É necessário um nível baixo de privilégio;
- Não é necessária interação com o Utilizador;
- Confidencialidade e Integridade não afetadas;
- Disponibilidade afetada com um grau de impacto alto.

CVE-2022-22638 Detail


Current Description

A null pointer dereference was addressed with improved validation. This issue is fixed in tvOS 15.4, iOS 15.4 and iPadOS 15.4, macOS Big Sur 11.6.5, Security Update 2022-003 Catalina, watchOS 8.5, macOS Monterey 12.3. An attacker in a privileged position may be able to perform a denial of service attack.

[View Analysis Description](#)

Severity CVSS Version 3.x CVSS Version 2.0

CVSS 3.x Severity and Metrics:

 NIST: NVD

Base Score: **6.5 MEDIUM**

Vector: CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:N/I:N/A:H

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

CVSS v3.1 Severity and Metrics:

Base Score: 6.5 MEDIUM

Vector: AV:N/AC:L/PR:L/UI:N/S:U/C:N/I:N/A:H

Impact Score: 3.6

Exploitability Score: 2.8

Attack Vector (AV): Network

Attack Complexity (AC): Low

Privileges Required (PR): Low

User Interaction (UI): None

Scope (S): Unchanged

Confidentiality (C): None

Integrity (I): None

Availability (A): High

Figura 30: Classificação CVE-2022-22638 plataforma NVD

Figura 31: Métricas CVE-2022-22638 plataforma NVD

6 Entidade Emissora

A Entidade Emissora é a entidade com poder de emitir e conferir autenticidade a um documento de identificação pessoal. É ainda responsável por providenciar mecanismos que garantem a autenticidade e integridade dos documentos digitais transmitidos tanto no modo *on-line*, quanto no modo *off-line*. Sendo assim, iremos neste momento perceber quais as vulnerabilidades que podemos encontrar nesta Entidade, bem como na sua infraestrutura.

Possuímos ainda alguma informação sobre a infraestrutura geral da Entidade Emissora, sendo a mesma a seguinte:

- Sistema operativo do servidor web: CentOS 7.8.2003
- Backend principal: Django v3.0
- Servidor web: UWSGI
- Base de dados principal: PostgreSQL 12.4
- Sistema operativo do serviço de gestão do sistema: Ubuntu 20.04
- Backend de gestão: Flask 1.0
- Servidor web: Gunicorn
- Base de dados de gestão: PostgreSQL 12.1 (a correr em um container Docker versão 19.03.6)

Esta infraestrutura encontra-se ainda numa fase de testes preliminares. Sendo assim, a mesma pode (e deve, caso necessário) ser alterada no futuro. Sendo assim, a equipa irá neste momento estudar cada uma das suas componentes mais importantes e perceber quais os riscos e vulnerabilidades que podemos ter nesta estrutura.

Nos casos em que não seja possível analisar uma vulnerabilidade através do **CVSS (Common Vulnerability Scoring System)**, iremos fazer a nossa própria análise (não havendo desta forma uma precisão tão grande).

6.1 Backend principal: Django v3.0

Iremos inicialmente estudar o backend principal de toda a estrutura da Entidade Emissora. Sendo assim, iremos estudar mais especificamente a framework web **Django v3.0**, identificando as principais vulnerabilidades.

6.1.1 CVE-2021-3281

A primeira vulnerabilidade encontrada foi publicada no **MITRE** no dia 22 de Janeiro de 2021, possuindo como CVE-ID, **CVE-2021-3281**. A vulnerabilidade encontrada incide num método do Django, mais propriamente no método que possui o nome *django.utils.archive.extract*, que permite a navegação de diretorias através de um arquivo que possui *paths* podendo estes ser absolutos ou parciais.

CVE-ID
CVE-2021-3281 Learn more at National Vulnerability Database (NVD) • CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information
Description
In Django 2.2 before 2.2.18, 3.0 before 3.0.12, and 3.1 before 3.1.6, the <code>django.utils.archive.extract</code> method (used by <code>"startapp --template"</code> and <code>"startproject --template"</code>) allows directory traversal via an archive with absolute paths or relative paths with dot segments.

Figura 32: Descrição CVE-2021-3281 na plataforma MITRE

Analisando agora as métricas, podemos perceber que esta vulnerabilidade possui um *Base Score* de 5.3, sendo considerada como uma vulnerabilidade de nível médio. Sabemos ainda que o seu impacto é de 1.4 e o *score* da sua *exploitability* é de 3.9. Quanto ao ataque, podemos perceber alguns aspetos pela informação recolhida:

- Tem de ser efetuado na Rede;
- É um ataque de complexidade baixa;
- Não é necessário qualquer tipo de privilégio ou interação com o Utilizador;
- Integridade afetada com um grau de impacto baixo.
- Confidencialidade e Disponibilidade não afetadas.

CVE-2021-3281 Detail


Current Description

In Django 2.2 before 2.2.18, 3.0 before 3.0.12, and 3.1 before 3.1.6, the `django.utils.archive.extract` method (used by `"startapp --template"` and `"startproject --template"`) allows directory traversal via an archive with absolute paths or relative paths with dot segments.

[+ View Analysis Description](#)

Severity CVSS Version 3.x CVSS Version 2.0

CVSS 3.x Severity and Metrics:

 **NIST: NVD** **Base Score: 5.3 MEDIUM** **Vector: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:L/A:N**

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

Figura 33: Classificação CVE-2021-3281 plataforma NVD

CVSS v3.1 Severity and Metrics:

Base Score: 5.3 MEDIUM

Vector: AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:L/A:N

Impact Score: 1.4

Exploitability Score: 3.9

Attack Vector (AV): Network

Attack Complexity (AC): Low

Privileges Required (PR): None

User Interaction (UI): None

Scope (S): Unchanged

Confidentiality (C): None

Integrity (I): Low

Availability (A): None

Figura 34: Métricas CVE-2021-3281 plataforma NVD

6.1.2 CVE-2020-7471

A segunda vulnerabilidade encontrada foi publicada no **MITRE** no dia 21 de Janeiro de 2020, possuindo como CVE-ID, **CVE-2020-7471**. A vulnerabilidade encontrada faz com que, se dados não confiáveis fossem utilizados como um delimitador *StringAgg*, o sistema ficava vulnerável. Mais especificamente, ao passar um delimitador criado adequadamente para uma instância *contrib.postgres.aggregates.StringAgg*, seria dessa forma possível injetar código SQL malicioso, comumente conhecido como um ato de **SQL-Injection**.

CVE-ID	
CVE-2020-7471	Learn more at National Vulnerability Database (NVD)
• CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information	
Description	
Django 1.11 before 1.11.28, 2.2 before 2.2.10, and 3.0 before 3.0.3 allows SQL Injection if untrusted data is used as a StringAgg delimiter (e.g., in Django applications that offer downloads of data as a series of rows with a user-specified column delimiter). By passing a suitably crafted delimiter to a contrib.postgres.aggregates.StringAgg instance, it was possible to break escaping and inject malicious SQL.	

Figura 35: Descrição CVE-2020-7471 na plataforma MITRE

Analisando agora as métricas, podemos perceber que esta vulnerabilidade possui um *Base Score* de 9.8, sendo considerada como uma vulnerabilidade de nível crítico. Sabemos ainda que o seu impacto é de 5.9 e o *score* da sua *exploitability* é de 3.9. Quanto ao ataque, podemos perceber alguns aspetos pela informação recolhida:

- Tem de ser efetuado na Rede;
- É um ataque de complexidade baixa;
- Não é necessário qualquer tipo de privilégio ou interação com o Utilizador;
- Confidencialidade, Integridade e Disponibilidade afetadas com um grau de impacto alto.

CVE-2020-7471 Detail

MODIFIED

This vulnerability has been modified since it was last analyzed by the NVD. It is awaiting reanalysis which may result in further changes to the information provided.


Current Description

Django 1.11 before 1.11.28, 2.2 before 2.2.10, and 3.0 before 3.0.3 allows SQL Injection if untrusted data is used as a StringAgg delimiter (e.g., in Django applications that offer downloads of data as a series of rows with a user-specified column delimiter). By passing a suitably crafted delimiter to a contrib.postgres.aggregates.StringAgg instance, it was possible to break escaping and inject malicious SQL.

[View Analysis Description](#)

Severity CVSS Version 3.x CVSS Version 2.0

CVSS 3.x Severity and Metrics:

 NIST: NVD

Base Score: **9.8 CRITICAL**

Vector: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

CVSS v3.1 Severity and Metrics:

Base Score: 9.8 CRITICAL

Vector: AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

Impact Score: 5.9

Exploitability Score: 3.9

Attack Vector (AV): Network

Attack Complexity (AC): Low

Privileges Required (PR): None

User Interaction (UI): None

Scope (S): Unchanged

Confidentiality (C): High

Integrity (I): High

Availability (A): High

Figura 36: Classificação CVE-2020-7471 plataforma NVD

Figura 37: Métricas CVE-2020-7471 plataforma NVD

6.1.3 CVE-2020-13254

A terceira vulnerabilidade encontrada foi publicada no **MITRE** no dia 21 de Maio de 2020, possuindo como CVE-ID, **CVE-2020-13254**. A vulnerabilidade encontrada indica que nos casos em que um *back-end memcached* não efetua *key validation*, a passagem de chaves de *cache* malformadas, pode resultar numa colisão de chaves e assim, num possível *leaking* de dados.

CVE-ID
CVE-2020-13254 Learn more at National Vulnerability Database (NVD)
• CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information
Description
An issue was discovered in Django 2.2 before 2.2.13 and 3.0 before 3.0.7. In cases where a memcached backend does not perform key validation, passing malformed cache keys could result in a key collision, and potential data leakage.

Figura 38: Descrição CVE-2020-13254 na plataforma MITRE

Analisando agora as métricas, podemos perceber que esta vulnerabilidade possui um *Base Score* de 5.9, sendo considerada como uma vulnerabilidade de nível médio. Sabemos ainda que o seu impacto é de 3.6 e o *score* da sua *exploitability* é de 2.2. Quanto ao ataque, podemos perceber alguns aspetos pela informação recolhida:

- Tem de ser efetuado na Rede;
- É um ataque de complexidade alta;
- Não é necessário qualquer tipo de privilégio ou interação com o Utilizador;
- Confidencialidade afetada com um grau de impacto alto.
- Integridade e Disponibilidade não afetadas.

CVE-2020-13254 Detail

MODIFIED

This vulnerability has been modified since it was last analyzed by the NVD. It is awaiting reanalysis which may result in further changes to the information provided.


Current Description

An issue was discovered in Django 2.2 before 2.2.13 and 3.0 before 3.0.7. In cases where a memcached backend does not perform key validation, passing malformed cache keys could result in a key collision, and potential data leakage.

[View Analysis Description](#)

Severity CVSS Version 3.x CVSS Version 2.0

CVSS 3.x Severity and Metrics:

 **NIST: NVD** **Base Score: 5.9 MEDIUM** **Vector: CVSS:3.1(AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:N/A:N)**

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

CVSS v3.1 Severity and Metrics:

Base Score: 5.9 MEDIUM
Vector: AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:N/A:N
Impact Score: 3.6
Exploitability Score: 2.2

Attack Vector (AV): Network
Attack Complexity (AC): High
Privileges Required (PR): None
User Interaction (UI): None
Scope (S): Unchanged
Confidentiality (C): High
Integrity (I): None
Availability (A): None

Figura 39: Classificação CVE-2020-13254 plataforma NVD

Figura 40: Métricas CVE-2020-13254 plataforma NVD

6.2 Sistema operativo do servidor web: CentOS 7.8.2003

Neste momento passamos a estudar o sistema operativo do servidor WEB, mais especificamente o Sistema operativo **CentOS 7.8.2003**, identificando as suas principais vulnerabilidades.

6.2.1 CVE-2020-5291

A primeira vulnerabilidade encontrada foi publicada no **MITRE** no dia 2 de Janeiro de 2020, possuindo como CVE-ID, **CVE-2020-5291**. A vulnerabilidade encontrada indica que o *Bubblewrap* (bwrap) antes da versão 0.4.1, se instalado no modo *setuid* e o kernel suportar *user namespaces* não privilegiados, então a opção 'bwrap --users2' pode ser utilizada para fazer o processo *setuid* continuar a rodar como root enquanto é rastreável. Isto, por sua vez, pode ser usado para obter permissões de root, fornecendo ao atacante permissões não desejáveis.

CVE-ID	
CVE-2020-5291	Learn more at National Vulnerability Database (NVD) • CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information
Description	
Bubblewrap (bwrap) before version 0.4.1, if installed in setuid mode and the kernel supports unprivileged user namespaces, then the 'bwrap --users2' option can be used to make the setuid process keep running as root while being traceable. This can in turn be used to gain root permissions. Note that this only affects the combination of bubblewrap in setuid mode (which is typically used when unprivileged user namespaces are not supported) and the support of unprivileged user namespaces. Known to be affected are: * Debian testing/unstable, if unprivileged user namespaces enabled (not default) * Debian buster-backports, if unprivileged user namespaces enabled (not default) * Arch if using 'linux-hardened', if unprivileged user namespaces enabled (not default) * Centos 7 Fatpak COPR, if unprivileged user namespaces enabled (not default) This has been fixed in the 0.4.1 release, and all affected users should update.	

Figura 41: Descrição CVE-2020-5291 na plataforma MITRE

Analisando agora as métricas, podemos perceber que esta vulnerabilidade possui um *Base Score* de 7.8, sendo considerada como uma vulnerabilidade de nível alto. Sabemos ainda que o seu impacto é de 5.9 e o *score* da sua *exploitability* é de 1.8. Quanto ao ataque, podemos perceber alguns aspetos pela informação recolhida:

- Tem de ser efetuado localmente;
- É um ataque de complexidade baixa;
- É necessário um nível de privilégio baixo;
- Não é necessária qualquer interação com o Utilizador;
- Confidencialidade, Integridade e Disponibilidade afetadas com um grau de impacto alto.

🚩 CVE-2020-5291 Detail

Current Description

Bubblewrap (bwrap) before version 0.4.1, if installed in setuid mode and the kernel supports unprivileged user namespaces, then the 'bwrap --users2' option can be used to make the setuid process keep running as root while being traceable. This can in turn be used to gain root permissions. Note that this only affects the combination of bubblewrap in setuid mode (which is typically used when unprivileged user namespaces are not supported) and the support of unprivileged user namespaces. Known to be affected are: * Debian testing/unstable, if unprivileged user namespaces enabled (not default) * Debian buster-backports, if unprivileged user namespaces enabled (not default) * Arch if using 'linux-hardened', if unprivileged user namespaces enabled (not default) * Centos 7 Fatpak COPR, if unprivileged user namespaces enabled (not default) This has been fixed in the 0.4.1 release, and all affected users should update.

[View Analysis Description](#)



Severity		CVSS Version 3.x	CVSS Version 2.0
CVSS 3.x Severity and Metrics:			
	NIST: NVD	Base Score: 7.2 HIGH	Vector: CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H
	CNA: GitHub, Inc.	Base Score: 7.2 HIGH	Vector: CVSS:3.1/AV:L/AC:H/PR:L/UI:R/S:C/C:H/I:H/A:N
<small>NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.</small>			
<small>Note: It is possible that the NVD CVSS may not match that of the CNA. The most common reason for this is that publicly available information does not provide sufficient detail or that information simply was not available at the time the CVSS vector string was assigned.</small>			

Figura 42: Classificação CVE-2020-5291 plataforma NVD

CVSS v3.1 Severity and Metrics:

Base Score: 7.8 HIGH
Vector: AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H
Impact Score: 5.9
Exploitability Score: 1.8

Attack Vector (AV): Local
Attack Complexity (AC): Low
Privileges Required (PR): Low
User Interaction (UI): None
Scope (S): Unchanged
Confidentiality (C): High
Integrity (I): High
Availability (A): High

Figura 43: Métricas CVE-2020-5291 plataforma NVD

6.2.2 CVE-2020-10230

A segunda vulnerabilidade encontrada foi publicada no **MITRE** no dia 8 de Março de 2020, possuindo como CVE-ID, **CVE-2020-10230**. A vulnerabilidade encontrada indica que o CentOS Web Panel permite ataques de SQL Injection através do parâmetro `/cwp_{SESSION_HASH}/admin/loader_ajax.php`.

CVE-ID
CVE-2020-10230 Learn more at National Vulnerability Database (NVD) • CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information
Description
CentOS-WebPanel.com (aka CWP) CentOS Web Panel (for CentOS 6 and 7) allows SQL Injection via the <code>/cwp_{SESSION_HASH}/admin/loader_ajax.php</code> term parameter.

Figura 44: Descrição CVE-2020-10230 na plataforma MITRE

Analisando agora as métricas, podemos perceber que esta vulnerabilidade possui um *Base Score* de 9.8, sendo considerada como uma vulnerabilidade de nível crítico. Sabemos ainda que o seu impacto é de 5.9 e o *score* da sua *exploitability* é de 3.9. Quanto ao ataque, podemos perceber alguns aspetos pela informação recolhida:

- Tem de ser efetuado na Rede;
- É um ataque de complexidade baixa;
- Não é necessário qualquer tipo de privilégio ou interação com o Utilizador;
- Confidencialidade, Integridade e Disponibilidade afetadas com um grau de impacto alto.

CVE-2020-10230 Detail


Current Description

CentOS-WebPanel.com (aka CWP) CentOS Web Panel (for CentOS 6 and 7) allows SQL Injection via the `/cwp_{SESSION_HASH}/admin/loader_ajax.php` term parameter.

[View Analysis Description](#)

Severity CVSS Version 3.x CVSS Version 2.0

CVSS 3.x Severity and Metrics:

 **NIST: NVD** **Base Score: 9.8 CRITICAL** **Vector: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H**

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

CVSS v3.1 Severity and Metrics:

Base Score: 9.8 CRITICAL

Vector: AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

Impact Score: 5.9

Exploitability Score: 3.9

Attack Vector (AV): Network

Attack Complexity (AC): Low

Privileges Required (PR): None

User Interaction (UI): None

Scope (S): Unchanged

Confidentiality (C): High

Integrity (I): High

Availability (A): High

Figura 45: Classificação CVE-2020-10230 plataforma NVD

Figura 46: Métricas CVE-2020-10230 plataforma NVD

6.3 Servidor web: UWSGI

Neste momento passamos a estudar o servidor WEB, mais especificamente o Servidor web **UWSGI**, identificando as suas principais vulnerabilidades.

6.3.1 CVE-2018-7490

A primeira vulnerabilidade encontrada foi publicada no **MITRE** no dia 26 de Fevereiro de 2018, possuindo como CVE-ID, **CVE-2018-7490**. A vulnerabilidade encontrada indica que este servidor Web manipula incorretamente uma verificação de *DOCUMENT_ROOT* quando é utilizada a opção *-php-docroot* que permite a travessia de diretórias.

CVE-ID
CVE-2020-10230 Learn more at National Vulnerability Database (NVD)
CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information
Description
CentOS-WebPanel.com (aka CWP) CentOS Web Panel (for CentOS 6 and 7) allows SQL Injection via the /cwp_(SESSION_HASH)/admin/loader_ajax.php term parameter.

Figura 47: Descrição CVE-2018-7490 na plataforma MITRE

Analisando agora as métricas, podemos perceber que esta vulnerabilidade possui um *Base Score* de 7.5, sendo considerada como uma vulnerabilidade de nível alto. Sabemos ainda que o seu impacto é de 3.6 e o *score* da sua *exploitability* é de 3.9. Quanto ao ataque, podemos perceber alguns aspetos pela informação recolhida:

- Tem de ser efetuado na Rede;
- É um ataque de complexidade baixa;
- Não é necessário qualquer tipo de privilégio ou interação com o Utilizador;
- Confidencialidade afetada com um grau de impacto alto.
- Integridade e Disponibilidade não afetadas.

CVE-2018-7490 Detail


Current Description

uWSGI before 2.0.17 mishandles a DOCUMENT_ROOT check during use of the `--php-docroot` option, allowing directory traversal.

[View Analysis Description](#)

Severity CVSS Version 3.x CVSS Version 2.0

CVSS 3.x Severity and Metrics:

 **NIST: NVD** **Base Score: 7.5 HIGH** **Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N**

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

CVSS v3.0 Severity and Metrics:

Base Score: 7.5 HIGH

Vector: AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N

Impact Score: 3.6

Exploitability Score: 3.9

Attack Vector (AV): Network

Attack Complexity (AC): Low

Privileges Required (PR): None

User Interaction (UI): None

Scope (S): Unchanged

Confidentiality (C): High

Integrity (I): None

Availability (A): None

Figura 48: Classificação CVE-2018-7490 plataforma NVD

Figura 49: Métricas CVE-2018-7490 plataforma NVD

6.3.2 CVE-2018-6758

A segunda vulnerabilidade encontrada foi publicada no **MITRE** no dia 6 de Fevereiro de 2018, possuindo como CVE-ID, **CVE-2018-6758**. A vulnerabilidade encontrada indica que a função *uwsgi_expand_path* contida em *core/utlis.c* possui um *stack-based buffer overflow*, através de um tamanho de diretoria elevado.

CVE-ID
CVE-2018-6758 Learn more at National Vulnerability Database (NVD) • CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information
Description
The uwsgi_expand_path function in core/utlis.c in Unbit uWSGI through 2.0.15 has a stack-based buffer overflow via a large directory length.

Figura 50: Descrição CVE-2018-6758 na plataforma MITRE

Analisando agora as métricas, podemos perceber que esta vulnerabilidade possui um *Base Score* de 9.8, sendo considerada como uma vulnerabilidade de nível crítico. Sabemos ainda que o seu impacto é de 5.9 e o *score* da sua *exploitability* é de 3.9. Quanto ao ataque, podemos perceber alguns aspetos pela informação recolhida:

- Tem de ser efetuado na Rede;
- É um ataque de complexidade baixa;
- Não é necessário qualquer tipo de privilégio ou interação com o Utilizador;
- Confidencialidade, Integridade e Disponibilidade afetadas com um grau de impacto alto.

CVE-2018-6758 Detail


Current Description

The uwsgi_expand_path function in core/utlis.c in Unbit uWSGI through 2.0.15 has a stack-based buffer overflow via a large directory length.

[View Analysis Description](#)

Severity CVSS Version 3.x CVSS Version 2.0

CVSS 3.x Severity and Metrics:

 NIST: NVD

Base Score: **9.8 CRITICAL**

Vector: CVSS:3.0(AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H)

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

CVSS v3.0 Severity and Metrics:

Base Score: 9.8 CRITICAL

Vector: AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

Impact Score: 5.9

Exploitability Score: 3.9

Attack Vector (AV): Network

Attack Complexity (AC): Low

Privileges Required (PR): None

User Interaction (UI): None

Scope (S): Unchanged

Confidentiality (C): High

Integrity (I): High

Availability (A): High

Figura 51: Classificação CVE-2018-6758 plataforma NVD

Figura 52: Métricas CVE-2018-6758 plataforma NVD

6.4 Base de dados principal: PostgreSQL 12.4

Neste momento passamos a estudar a base de dados principal, mais especificamente a base de dados que utiliza **PostgreSQL 12.4**, identificando as suas principais vulnerabilidades.

6.4.1 CVE-2021-32029

A primeira vulnerabilidade encontrada foi publicada no **MITRE** no dia 4 de Maio de 2021, possuindo como CVE-ID, **CVE-2021-32029**. A vulnerabilidade encontrada indica que utilizando um comando "UPDATE ... RETURNING" numa tabela criada para fins específicos, um Utilizador autenticado podia ler *bytes* arbitrários de dados da memória do servidor.

CVE-ID
CVE-2021-32029 Learn more at National Vulnerability Database (NVD)
• CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information
Description
A flaw was found in postgresql. Using an UPDATE ... RETURNING command on a purpose-crafted table, an authenticated database user could read arbitrary bytes of server memory. The highest threat from this vulnerability is to data confidentiality.

Figura 53: Descrição CVE-2021-32029 na plataforma MITRE

Analisando agora as métricas, podemos perceber que esta vulnerabilidade possui um *Base Score* de 6.5, sendo considerada como uma vulnerabilidade de nível médio. Sabemos ainda que o seu impacto é de 3.6 e o *score* da sua *exploitability* é de 2.8. Quanto ao ataque, podemos perceber alguns aspetos pela informação recolhida:

- Tem de ser efetuado na Rede;
- É um ataque de complexidade baixa;
- É necessário um nível de privilégios baixo;
- Não é necessária qualquer interação com o Utilizador;
- Confidencialidade afetada com um grau de impacto alto.
- Integridade e Disponibilidade não afetadas.

🚩 CVE-2021-32029 Detail


Current Description

A flaw was found in postgresql. Using an UPDATE ... RETURNING command on a purpose-crafted table, an authenticated database user could read arbitrary bytes of server memory. The highest threat from this vulnerability is to data confidentiality.

[View Analysis Description](#)

Severity CVSS Version 3.x CVSS Version 2.0

CVSS 3.x Severity and Metrics:

 **NIST: NVD** **Base Score: 6.5 MEDIUM** **Vector: CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N**

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

CVSS v3.1 Severity and Metrics:

Base Score: 6.5 MEDIUM
Vector: AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N
Impact Score: 3.6
Exploitability Score: 2.8

Attack Vector (AV): Network
Attack Complexity (AC): Low
Privileges Required (PR): Low
User Interaction (UI): None
Scope (S): Unchanged
Confidentiality (C): High
Integrity (I): None
Availability (A): None

Figura 54: Classificação CVE-2021-32029 plataforma NVD

Figura 55: Métricas CVE-2021-32029 plataforma NVD

6.4.2 CVE-2020-25695

A primeira vulnerabilidade encontrada foi publicada no **MITRE** no dia 16 de Setembro de 2020, possuindo como CVE-ID, **CVE-2020-25695**. A vulnerabilidade encontrada indica que um atacante com permissões para criar objetos não temporários em pelo menos um *schema* da base de dados, pode executar funções SQL arbitrárias sob a entidade de um *superuser*.

CVE-ID	
CVE-2020-25695	Learn more at National Vulnerability Database (NVD)
• CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information	
Description	
A flaw was found in PostgreSQL versions before 13.1, before 12.5, before 11.10, before 10.15, before 9.6.20 and before 9.5.24. An attacker having permission to create non-temporary objects in at least one schema can execute arbitrary SQL functions under the identity of a superuser. The highest threat from this vulnerability is to data confidentiality and integrity as well as system availability.	

Figura 56: Descrição CVE-2020-25695 na plataforma MITRE

Analisando agora as métricas, podemos perceber que esta vulnerabilidade possui um *Base Score* de 8.8, sendo considerada como uma vulnerabilidade de nível alto. Sabemos ainda que o seu impacto é de 5.9 e o *score* da sua *exploitability* é de 2.8. Quanto ao ataque, podemos perceber alguns aspetos pela informação recolhida:

- Tem de ser efetuado na Rede;
- É um ataque de complexidade baixa;
- É necessário um nível de privilégios baixo;
- Não é necessária qualquer interação com o Utilizador;
- Confidencialidade, Integridade e Disponibilidade afetadas com um grau de impacto alto.

CVE-2020-25695 Detail

MODIFIED

This vulnerability has been modified since it was last analyzed by the NVD. It is awaiting reanalysis which may result in further changes to the information provided.

Current Description

A flaw was found in PostgreSQL versions before 13.1, before 12.5, before 11.10, before 10.15, before 9.6.20 and before 9.5.24. An attacker having permission to create non-temporary objects in at least one schema can execute arbitrary SQL functions under the identity of a superuser. The highest threat from this vulnerability is to data confidentiality and integrity as well as system availability.

[View Analysis Description](#)

Severity

CVSS Version 3.x CVSS Version 2.0

CVSS 3.x Severity and Metrics:

NIST: NVD

Base Score: 8.8 HIGH

Vector: CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

CVSS v3.1 Severity and Metrics:

Base Score: 8.8 HIGH

Vector: AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H

Impact Score: 5.9

Exploitability Score: 2.8

Attack Vector (AV): Network

Attack Complexity (AC): Low

Privileges Required (PR): Low

User Interaction (UI): None

Scope (S): Unchanged

Confidentiality (C): High

Integrity (I): High

Availability (A): High

Figura 57: Classificação CVE-2020-25695 plataforma NVD

Figura 58: Métricas CVE-2020-25695 plataforma NVD

6.5 Backend de gestão: Flask 1.0

Neste momento passamos a estudar o backend da gestão, mais especificamente a framework **Flask 1.0**, identificando as suas principais vulnerabilidades.

6.5.1 CVE-2022-24880

A primeira vulnerabilidade encontrada foi publicada no **MITRE** no dia 10 de Fevereiro de 2022, possuindo como CVE-ID, **CVE-2022-24880**. A vulnerabilidade encontrada indica que *flask-session-captcha* é um pacote que permite aos Utilizadores estender o Flask ao adicionar um *image based captcha* armazenado numa sessão *server side*. A função `captcha.validate()` iria retornar `None` se não fosse passado nenhum valor. Se os Utilizadores verificassem se o valor retornado era `False`, a verificação *captcha* poderia ser ignorada.

CVE-ID	
CVE-2022-24880	Learn more at National Vulnerability Database (NVD)
• CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information	
Description	
flask-session-captcha is a package which allows users to extend Flask by adding an image based captcha stored in a server side session. In versions prior to 1.2.1, the <code>captcha.validate()</code> function would return <code>None</code> if passed no value (e.g. by submitting an having an empty form). If implementing users were checking the return value to be <code>False</code> , the captcha verification check could be bypassed. Version 1.2.1 fixes the issue. Users can workaround the issue by not explicitly checking that the value is <code>False</code> . Checking the return value less explicitly should still work.	

Figura 59: Descrição CVE-2022-24880 na plataforma MITRE

Analisando agora as métricas, podemos perceber que esta vulnerabilidade possui um *Base Score* de 5.3, sendo considerada como uma vulnerabilidade de nível médio. Sabemos ainda que o seu impacto é de 1.4 e o *score* da sua *exploitability* é de 3.9. Quanto ao ataque, podemos perceber alguns aspetos pela informação recolhida:

- Tem de ser efetuado na Rede;
- É um ataque de complexidade baixa;
- Não é necessário qualquer tipo de privilégio ou interação com o Utilizador;
- Integridade afetada com um grau de impacto baixo.
- Confidencialidade e Disponibilidade não afetadas.

🚩 CVE-2022-24880 Detail

AWAITING ANALYSIS


This vulnerability is currently awaiting analysis.

Description

flask-session-captcha is a package which allows users to extend Flask by adding an image based captcha stored in a server side session. In versions prior to 1.2.1, the `captcha.validate()` function would return `None` if passed no value (e.g. by submitting an having an empty form). If implementing users were checking the return value to be `False`, the captcha verification check could be bypassed. Version 1.2.1 fixes the issue. Users can workaround the issue by not explicitly checking that the value is `False`. Checking the return value less explicitly should still work.

Severity CVSS Version 3.x CVSS Version 2.0

CVSS 3.x Severity and Metrics:

 NIST: NVD

Base Score: **N/A**

NVD score not yet provided.

 CNA: GitHub, Inc.

Base Score: **5.3 MEDIUM**

Vector: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:L/A:N

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have not published a CVSS score for this CVE at this time. NVD Analysts use publicly available information at the time of analysis to associate CVSS vector strings. A CNA provided score within the CVE List has been displayed.

CVSS v3.1 Severity and Metrics:

Base Score: 5.3 MEDIUM

Vector: AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:L/A:N

Impact Score: 1.4

Exploitability Score: 3.9

Attack Vector (AV): Network

Attack Complexity (AC): Low

Privileges Required (PR): None

User Interaction (UI): None

Scope (S): Unchanged

Confidentiality (C): None

Integrity (I): Low

Availability (A): None

Figura 60: Classificação CVE-2022-24880 plataforma NVD

Figura 61: Métricas CVE-2022-24880 plataforma NVD

6.5.2 CVE-2018-16516

A primeira vulnerabilidade encontrada foi publicada no **MITRE** no dia 5 de Setembro de 2018, possuindo como CVE-ID, **CVE-2018-16516**. A vulnerabilidade encontrada indica que o ficheiro *helper.py* possui Reflected XSS através de um URL criado.

CVE-ID
CVE-2018-16516 Learn more at National Vulnerability Database (NVD)
CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information
Description
helpers.py in Flask-Admin 1.5.2 has Reflected XSS via a crafted URL.

Figura 62: Descrição CVE-2018-16516 na plataforma MITRE

Analisando agora as métricas, podemos perceber que esta vulnerabilidade possui um *Base Score* de 6.1, sendo considerada como uma vulnerabilidade de nível médio. Sabemos ainda que o seu impacto é de 2.7 e o *score* da sua *exploitability* é de 2.8. Quanto ao ataque, podemos perceber alguns aspetos pela informação recolhida:

- Tem de ser efetuado na Rede;
- É um ataque de complexidade baixa;
- Não é necessário qualquer tipo de privilégio;
- É necessária interação com o Utilizador;
- Confidencialidade e Integridade afetadas com um grau de impacto baixo.
- Disponibilidade não afetada.

CVE-2018-16516 Detail

MODIFIED

This vulnerability has been modified since it was last analyzed by the NVD. It is awaiting reanalysis which may result in further changes to the information provided.

Current Description

helpers.py in Flask-Admin 1.5.2 has Reflected XSS via a crafted URL.

[View Analysis Description](#)

Severity

CVSS Version 3.x CVSS Version 2.0

CVSS 3.x Severity and Metrics:



NIST: NVD

Base Score: **6.1 MEDIUM**

Vector: CVSS:3.0/(AV:N)/(AC:L)/(PR:N)/(UI:R)/(S:C)/(C:L)/(I:L)/(A:N)

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

CVSS v3.0 Severity and Metrics:

Base Score: 6.1 MEDIUM

Vector: AV:N/AC:L/PR:N/UI:R/S:C/C:L/I:L/A:N

Impact Score: 2.7

Exploitability Score: 2.8

Attack Vector (AV): Network

Attack Complexity (AC): Low

Privileges Required (PR): None

User Interaction (UI): Required

Scope (S): Changed

Confidentiality (C): Low

Integrity (I): Low

Availability (A): None

Figura 63: Classificação CVE-2018-16516 plataforma NVD

Figura 64: Métricas CVE-2018-16516 plataforma NVD

7 Relações entre Entidades

Neste momento será então necessário perceber quais as vulnerabilidades que podem ocorrer na comunicação entre as varias entidades do sistema total. Assim, será estudada cada uma destas interações de modo a perceber em que ponto é que as mesmas podem falhar e quais as situações em que são mais precárias. Sabemos então que possuímos 3 interações diferentes:

- Portador - Entidade Emissora
- Verificador/Leitor - Entidade Emissora
- Portador - Verificador/Leitor

Mais uma vez, nos casos em que não seja possível analisar uma vulnerabilidade através do **CVSS (Common Vulnerability Scoring System)**, iremos fazer a nossa própria análise (não havendo desta forma uma precisão tão grande).

7.1 Portador - Entidade Emissora

Tal como foi referido anteriormente, sabemos que, na primeira vez que o portador usa a aplicação, este conecta-se com a infra-estrutura de uma entidade emissora de documentos (usando comunicação TCP/IP) de forma a fazer download de todos os dados associados a esse documento. Para isso, o cidadão autentica-se no respetivo serviço por forma a iniciar a transferência do seu documento. Periodicamente, esta operação é repetida, de modo aos dados estarem sempre o mais atualizados possível. Por fim, sabemos ainda que estes dados são transferidos sobre o formato JSON.

Tendo em conta estas afirmações, iremos então passar a estudar vulnerabilidades existentes nesta interação, descrevendo as mesmas.

7.1.1 Attack Man-In-The-Middle

Um dos ataques possíveis de acontecer numa comunicação que utiliza o protocolo TCP/IP é o famoso ataque Man in the Middle. Este ataque acontece quando uma comunicação entre dois sistemas é intercetada por uma entidade externa. Os dados que são transmitidos de fim a fim nesta comunicação serão também eles intercetados.

Neste tipo de ataque podemos perceber que o *attacker* irá passar a controlar o fluxo da comunicação, e pode eliminar ou alterar a informação enviada por uma das entidades originais sem qualquer perceção, neste caso, do Portador ou da Entidade Emissora.

O ataque *man-in-the-middle* é então, uma forma de ataque em que os dados trocados entre duas partes são de alguma forma interceptados, registrados e, possivelmente, alterados pelo atacante sem que as vitimas se apercebam. Assim, um dos sistemas poderá estar a receber informações completamente diferentes das iniciais, ou até mesmo, ser bloqueada a comunicação.

Analisando a matriz de análise de risco do Anexo A (Figura 74), podemos determinar a "*likelihood*" deste ataque como **Probable** e a "*severity*" como **Undesirable**, sendo desta forma um ataque com um risco de impacto alto, onde iremos precisar de procurar ajuda para resolver o mesmo.

A maneira mais forte de prevenir ataques deste tipo, é possuir uma encriptação bastante elevada e bem construída em toda a comunicação e interação das entidades.

7.1.2 IP spoofing

Um outro ataque do tipo Man-in-the-Middle é conhecido como **IP spoofing**. Este é uma especificação do ataque Man-in-the-Middle e por isso o seu conceito é análogo ao explicado anteriormente. No entanto, este distingue-se por uma particularidade diferente, que o torna muito mais forte que o ataque anterior.

Como sabemos, os *attackers* comumente denominados de *hackers*, utilizam a falsificação de endereços IP para realizarem muitos dos seus ataques, como por exemplo a iniciação de ataques DDoS, no entanto, esta técnica também pode ser utilizada para efetuar alguns ataques Man-in-the-Middle.

Neste tipo de ataque Man-in-the-Middle denominado **IP spoofing**, o attacker irá ficar entre as duas partes comunicantes e irá falsificar cada um dos seus endereços IP para o outro. Desta forma, o que irá acontecer é que os dois sistemas comunicantes não irão enviar os seus pacotes para o destino final, mas sim para o attacker, que se encontra escondido no centro da comunicação. Este passará então a ter um controlo absoluto do fluxo de comunicação.

Analisando a matriz de análise de risco do Anexo A (Figura 74), podemos determinar a "*likelihood*" deste ataque como **Probable** e a "*severity*" como **Undesirable**, sendo desta forma um ataque com um risco de impacto alto, onde iremos precisar de procurar ajuda para resolver o mesmo.

A maneira mais forte de prevenir ataques deste tipo, é possuir uma encriptação bastante elevada e bem construída em toda a comunicação e interação das duas entidades.

7.1.3 JSON Hijacking

Neste tipo de ataque, um invasor tem como alvo um sistema que usa **JavaScript Object Notation (JSON)** como um mecanismo de transporte entre o cliente e o servidor (comum em sistemas Web 2.0) para roubar informações possivelmente confidenciais transmitidas do servidor de volta ao cliente dentro do objeto JSON aproveitando-se de uma brecha na Política do navegador que não proíbe que o JavaScript de um site seja incluído e executado no contexto de outro site.

Desta forma, um *attacker* poderia aceder a informações da mID que é enviada ao Portador através do formato JSON (JavaScript Object Notation), roubando informações sobre a mesma.

Analisando a matriz de análise de risco do Anexo A (Figura 74), podemos determinar a "*likelihood*" deste ataque como **Improbable** e a "*severity*" como **Undesirable**, sendo desta forma um ataque com um risco de impacto médio, onde será preciso esforço para resolver este erro.

A maneira mais forte de prevenir ataques deste tipo, é possuir uma encriptação bastante elevada e bem construída em toda a comunicação e interação das duas entidades.

7.1.4 Falta de Autenticação

Como podemos perceber pela introdução deste tópico, quando um portador usa a aplicação mID pela primeira vez, este conecta-se com a infraestrutura de uma entidade emissora de documentos, para efetuar o download de todos os dados associados a este documento. Sabemos ainda que, para isso, o cidadão autentica-se no respetivo serviço por forma a iniciar a transferência do seu documento. Todo este processo é acertivo, no entanto, é apenas depois dele que podemos encontrar uma vulnerabilidade.

É também sabido que, um Utilizador irá periodicamente repetir esta operação para atualizar os seus dados e possuir sempre a informação mais recente possível. No entanto, nesta atualização de dados, o Utilizador não irá necessitar de recorrer a uma autenticação explícita ao Sistema. Deste modo, o sistema e o processo tornam-se vulneráveis a ataques.

Estes ataques terão mais impacto no mundo "real" do que no digital. Imaginando que o dispositivo, ou seja, o *smartphone*, de um portador é roubado ou perdido, e não possua qualquer tipo de password para o seu manuseamento, todo o processo de autenticação estará em causa, dado não ser preciso realizar uma autenticação explícita.

Analisando a matriz de análise de risco do Anexo A (Figura 74), podemos determinar a "*likelihood*" deste ataque como **Improbable** e a "*severity*" como **Undesirable**, sendo desta forma um ataque com um risco de impacto médio, onde será preciso esforço para resolver este erro.

A maneira mais forte de prevenir ataques deste tipo, é possuir algum tipo de mecanismo que force o portador a autenticar-se todas as vezes que necessita de uma atualização ou de troca de informações com a Entidade Emissora.

7.2 Verificador/Leitor - Entidade Emissora

Tal como a comunicação feita entre o Portador e a Entidade Emissora, também a comunicação entre o Verificador/Leitor e a Entidade Emissora é feita através de protocolos **TPC/IP**. Nesta comunicação irão ser transmitidos dados que serão passíveis de verificar a autenticidade do Portador e do seu documento mID.

Deste forma, e como a comunicação é análogo à já falada anteriormente, podemos perceber, que também os problemas nesta comunicação serão os mesmos. Teremos então, como vulnerabilidades desta comunicação:

- Attack Man-In-The-Middle
- IP spoofing
- JSON Hijacking

7.3 Portador - Verificador/Leitor

A comunicação que se estudará agora, é a interação principal do nosso sistema. É através desta interação entre o dispositivo Portador e o dispositivo Verificador/Leitor que será feita a verificação do mID. Esta comunicação, pode ser feita de duas formas:

- **off-line** - dispositivo do portador transfere os atributos de identificação diretamente para o dispositivo leitor, assim como os dados necessários para a sua verificação;
- **on-line** - dispositivo leitor transfere um token de autorização para que o verificador consulte diretamente a entidade emissora do documento.

Para além dos tipos possíveis de comunicação, sabemos ainda que o estabelecimento da comunicação é iniciada sempre pelo dispositivo do Portador através de um **QR Code** que contem toda a informação necessária para que o dispositivo Verificador/Leitor o encontre e inicie a conexão, que pode usar uma das seguintes tecnologias:

- **BLE - Bluetooth Low Energy;**
- **NFC - Near Field Communication;**
- **WiFi-Aware;**

Sabemos ainda, por fim, que a comunicação é suportada por mensagens codificados no formato **CBOR - Concise Binary Object Representation**.

Tendo em conta todas estas informações, iremos então estudar possíveis vulnerabilidades que podem ocorrer nesta interação/comunicação.

7.3.1 BLE - Bluetooth Low Energy

Neste momento passamos a estudar a tecnologia **Bluetooth Low Energy**, identificando as suas principais vulnerabilidades.

CVE-2019-19196

A primeira vulnerabilidade encontrada foi publicada no **MITRE** no dia 21 de Novembro de 2019, possuindo como CVE-ID, **CVE-2019-19196**. A vulnerabilidade encontrada indica que é aceite uma solicitação de emparelhamento com um tamanho de chave maior que 16 bytes, permitindo a que um invasor no alcance de rádio cause um *buffer-overflow* e um *denial of service* (crash) através de pacotes por si criados.

CVE-ID
CVE-2019-19196 Learn more at National Vulnerability Database (NVD) • CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information
Description
The Bluetooth Low Energy Secure Manager Protocol (SMP) implementation on Telink Semiconductor BLE SDK versions before November 2019 for TLSR825x through 3.4.0, TLSR823x through 1.3.0, and TLSR826x through 3.3 devices accepts a pairing request with a key size greater than 16 bytes, allowing an attacker in radio range to cause a buffer overflow and denial of service (crash) via crafted packets.

Figura 65: Descrição CVE-2019-19196 na plataforma MITRE

Analisando agora as métricas, podemos perceber que esta vulnerabilidade possui um *Base Score* de 6.5, sendo considerada como uma vulnerabilidade de nível médio. Sabemos ainda que o seu impacto é de 3.6 e o *score* da sua *exploitability* é de 2.8. Quanto ao ataque, podemos perceber alguns aspetos pela informação recolhida:

- Tem de ser efetuado adjacientemente;
- É um ataque de complexidade baixa;
- Não é necessário qualquer tipo de privilégio ou interação com o Utilizador;
- Confidencialidade e Integridade não afetadas;
- Disponibilidade afetada com um grau de impacto alto.

CVE-2019-19196 Detail


Current Description

The Bluetooth Low Energy Secure Manager Protocol (SMP) implementation on Telink Semiconductor BLE SDK versions before November 2019 for TLSR825x through 3.4.0, TLSR823x through 1.3.0, and TLSR826x through 3.3 devices accepts a pairing request with a key size greater than 16 bytes, allowing an attacker in radio range to cause a buffer overflow and denial of service (crash) via crafted packets.

[View Analysis Description](#)

Severity CVSS Version 3.x CVSS Version 2.0

CVSS 3.x Severity and Metrics:

 NIST: NVD

Base Score: **6.5 MEDIUM**

Vector: CVSS:3.1/(AV:A)/(AC:L)/(PR:N)/(UI:N)/(S:U)/(C:N)/(I:N)/(A:H)

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

CVSS v3.1 Severity and Metrics:

Base Score: 6.5 MEDIUM

Vector: AV:A/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:H

Impact Score: 3.6

Exploitability Score: 2.8

Attack Vector (AV): Adjacent

Attack Complexity (AC): Low

Privileges Required (PR): None

User Interaction (UI): None

Scope (S): Unchanged

Confidentiality (C): None

Integrity (I): None

Availability (A): High

Figura 66: Classificação CVE-2019-19196 plataforma NVD

Figura 67: Métricas CVE-2019-19196 plataforma NVD

CVE-2020-10061

A segunda vulnerabilidade encontrada foi publicada no **MITRE** no dia 4 de Março de 2020, possuindo como CVE-ID, **CVE-2020-10061**. A vulnerabilidade encontrada indica que o manuseamento inadequado de casos específicos de *buffers* cheios pode resultar em corrupção de memória.

CVE-ID
CVE-2020-10061 Learn more at National Vulnerability Database (NVD)
CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information
Description
Improper handling of the full-buffer case in the Zephyr Bluetooth implementation can result in memory corruption. This issue affects: zephyrproject-rtos zephyr version 2.2.0 and later versions, and version 1.14.0 and later versions.

Figura 68: Descrição CVE-2020-10061 na plataforma MITRE

Analisando agora as métricas, podemos perceber que esta vulnerabilidade possui um *Base Score* de 8.8, sendo considerada como uma vulnerabilidade de nível alto. Sabemos ainda que o seu impacto é de 5.9 e o *score* da sua *exploitability* é de 2.8. Quanto ao ataque, podemos perceber alguns aspetos pela informação recolhida:

- Tem de ser efetuado adjacientemente;
- É um ataque de complexidade baixa;
- Não é necessário qualquer tipo de privilégio ou interação com o Utilizador;
- Confidencialidade, Integridade e Disponibilidade afetadas com um grau de impacto alto.

CVE-2020-10061 Detail

Current Description

Improper handling of the full-buffer case in the Zephyr Bluetooth implementation can result in memory corruption. This issue affects: zephyrproject-rtos zephyr version 2.2.0 and later versions, and version 1.14.0 and later versions.


[View Analysis Description](#)

Severity

CVSS Version 3.x


CVSS Version 2.0

CVSS 3.x Severity and Metrics:

 NIST: NVD

Base Score: **8.8 HIGH**

Vector: CVSS-3.1/AV:A/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

 CNA: Zephyr Project

Base Score: **8.1 HIGH**

Vector: CVSS-3.1/AV:L/AC:H/PR:N/UI:N/S:C/C:H/I:H/A:H

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: It is possible that the NVD CVSS may not match that of the CNA. The most common reason for this is that publicly available information does not provide sufficient detail or that information simply was not available at the time the CVSS vector string was assigned.

CVSS v3.1 Severity and Metrics:

Base Score: 8.8 HIGH

Vector: AV:A/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

Impact Score: 5.9

Exploitability Score: 2.8

Attack Vector (AV): Adjacent

Attack Complexity (AC): Low

Privileges Required (PR): None

User Interaction (UI): None

Scope (S): Unchanged

Confidentiality (C): High

Integrity (I): High

Availability (A): High

Figura 69: Classificação CVE-2020-10061 plataforma NVD

Figura 70: Métricas CVE-2020-10061 plataforma NVD

7.3.2 NFC - Near Field Communication

Neste momento passamos a estudar a tecnologia **Near Field Communication**, identificando as suas principais vulnerabilidades.

Eavesdropping

Num cenário de **Eavesdropping** o invasor (*attacker*) usa uma antena para registar a comunicação entre os dispositivos NFC. Apesar da comunicação NFC ocorrer entre dispositivos que se encontram próximos um do outro, este tipo de ataque ainda é viável. A interceptação de uma troca de dados através de NFC nem sempre se traduz num roubo de informações. Em alguns casos, o ataque visa apenas corromper as informações trocadas, tornando-as inúteis e a comunicação improdutiva.

Analisando a matriz de análise de risco do Anexo A (Figura 74), podemos determinar a "*likelihood*" deste ataque como **Improbable** e a "*severity*" como **Undesirable**, sendo desta forma um ataque com um risco de impacto médio, onde será preciso esforço para resolver este erro.

A maneira mais forte de prevenir ataques deste tipo, é utilizar um canal seguro que deve ser estabelecido entre os dispositivos NFC, implementando nele métodos de criptografia que fortaleçam a comunicação. Para além desta, a proximidade das unidades de comunicação é também outra forma de prevenção para a realização destes ataques, mas não elimina os riscos.

Relay attack

Num cenário de **Relay attack**, é explorada a conformidade do protocolo ISO/IEC 14443 da NFC. Neste tipo de ataque o invasor (*attacker*) deve encaminhar o *request* do leitor para a vítima e retransmitir a sua resposta ao leitor em tempo real para realizar uma tarefa completa, de modo a fingir ser o proprietário, no nosso caso, da mID.

Analisando a matriz de análise de risco do Anexo A (Figura 74), podemos determinar a "*likelihood*" deste ataque como **Improbable** e a "*severity*" como **Undesirable**, sendo desta forma um ataque com um risco de impacto médio, onde será preciso esforço para resolver este erro.

A maneira mais forte de prevenir ataques deste tipo, é adotar a medida mais simples, que consiste em blindar o cartão ao lado do usuário com uma caixa chamada gaiola de Faraday (Faraday cages). No entanto, esta não será a mais oportuna, e por isso uma solução mais viável será adotar protocolos de limitação de distância do sistema **RFID (Identificação por radiofrequência)** para que o leitor saiba se o cartão está apresentado dentro do recetáculo eletromagnético, ou se está a ser realizado um Relay attack.

7.3.3 CBOR - Concise Binary Object Representation

Neste momento passamos a estudar o tipo de codificação utilizada na comunicação, mais especificamente **CBOR - Concise Binary Object Representation**, identificando as suas principais vulnerabilidades.

CVE-2020-24753

A vulnerabilidade encontrada foi publicada no **MITRE** no dia 28 de Agosto de 2020, possuindo como CVE-ID, **CVE-2020-24753**. A vulnerabilidade encontrada indica que pode haver uma corrupção de memória em *Objective Open CBOR Run-time*, que podem permitir que um *attacker* execute código arbitrário através de um *input* CBOR construído com recurso ao decodificador **cbor2json**. Um erro não detetado ao decodificar strings de texto **CBOR Major Type 3** leva ao uso de um valor da *stack* não inicializado controlável pelo *attacker*. Isso pode ser usado para modificar a memória, causando uma falha ou ainda ser usado para corrupção da *heap*.

CVE-ID
CVE-2020-24753 Learn more at National Vulnerability Database (NVD)
CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information
Description
A memory corruption vulnerability in Objective Open CBOR Run-time (oocborrt) in versions before 2020-08-12 could allow an attacker to execute code via crafted Concise Binary Object Representation (CBOR) input to the cbor2json decoder. An uncaught error while decoding CBOR Major Type 3 text strings leads to the use of an attacker-controllable uninitialized stack value. This can be used to modify memory, causing a crash or potentially exploitable heap corruption.

Figura 71: Descrição CVE-2020-24753 na plataforma MITRE

Analisando agora as métricas, podemos perceber que esta vulnerabilidade possui um *Base Score* de 9.8, sendo considerada como uma vulnerabilidade de nível crítico. Sabemos ainda que o seu impacto é de 5.9 e o *score* da sua *exploitability* é de 3.9. Quanto ao ataque, podemos perceber alguns aspetos pela informação recolhida:

- Tem de ser efetuado na Rede;
- É um ataque de complexidade baixa;
- Não é necessário qualquer tipo de privilégio ou interação com o Utilizador;
- Confidencialidade, Integridade e Disponibilidade afetadas com um grau de impacto alto.

CVE-2020-24753 Detail

Current Description

A memory corruption vulnerability in Objective Open CBOR Run-time (oocborrt) in versions before 2020-08-12 could allow an attacker to execute code via crafted Concise Binary Object Representation (CBOR) input to the cbor2json decoder. An uncaught error while decoding CBOR Major Type 3 text strings leads to the use of an attacker-controllable uninitialized stack value. This can be used to modify memory, causing a crash or potentially exploitable heap corruption.


[+View Analysis Description](#)

Severity

CVSS Version 3.x

CVSS Version 2.0

CVSS 3.x Severity and Metrics:

 NIST: NVD

Base Score: **9.8 CRITICAL**

Vector: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

CVSS v3.1 Severity and Metrics:

Base Score: 9.8 CRITICAL

Vector: AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

Impact Score: 5.9

Exploitability Score: 3.9

Attack Vector (AV): Network
Attack Complexity (AC): Low
Privileges Required (PR): None
User Interaction (UI): None
Scope (S): Unchanged
Confidentiality (C): High
Integrity (I): High
Availability (A): High

Figura 72: Classificação CVE-2020-24753 plataforma NVD

Figura 73: Métricas CVE-2020-24753 plataforma NVD

7.4 Excesso de Informação

Uma outra vulnerabilidade que podemos encontrar presente neste tipo de comunicação é o facto de por vezes poder haver excesso de informação e de dados trocados durante toda a comunicação. No caso em estudo, o dispositivo leitor poderá pedir mais informação do que aquela que é necessária para uma dada transação de dados. Apesar de, na outra ponta da comunicação, o Portador ter a possibilidade de decidir se pretende aceitar a transferência da totalidade dos atributos pedidos, ou apenas de um sub-conjunto, este poderá ser induzido em erro. Muitas vezes podemos assinalar que aceitamos transferir informação que não era necessária, quer por falta de atenção, tempo (e por isso Portador está com pressa), ou ainda porque este atributo deu "*blend-in*" com os outros e não nos apercebemos do mesmo.

Analisando a matriz de análise de risco do Anexo A (Figura 74), podemos determinar a "*likelihood*" deste ataque como **Possible** e a "*severity*" como **Acceptable**, sendo desta forma um ataque com um risco de impacto baixo, que poderá ser facilmente resolvido.

A maneira mais forte de prevenir ataques deste tipo, é o Portador decidir sempre com cuidado e atenção quais os atributos que deseja fazer a transferência, de maneira a evitar da melhor forma o erro humano. Para além disso podem ainda ser criados mecanismos de auto-seleção de atributos, de modo a não escolhermos os desnecessários.

8 Recomendações de Segurança

Neste momento iremos então apresentar algumas soluções e recomendações de segurança que acreditamos que o Sistema deva possuir.

8.1 Login e Código de acesso

Como sabemos, a maioria das aplicações dos dias de hoje possuem o processo de Login e autenticação. Neste projeto, não devemos deixar de seguir a norma e implementar então esta funcionalidade. Como será de esperar, esta funcionalidade é já implícita, dado que cada pessoa irá possuir credenciais diferentes e por isso a aplicação deve apenas apresentar os seus dados individuais (para cada pessoa irá apresentar informação diferente).

O problema é que, a maioria das aplicações de hoje em dia, devido à conectividade com os dados armazenados na sua conta Google, ou no seu dispositivo, pode abrir uma aplicação e inserir os dados de Login com apenas um clique, dado que estes já se encontram previamente guardados no dispositivo. A praticidade desta função, é particularmente boa, dado que retira o esforço e chatice de estar sempre a colocar a password e e-mail quando entramos numa aplicação. No entanto, esta funcionalidade peca quando, por exemplo, um dispositivo é **roubado** ou **perdido** e não possui PIN de desbloqueio. Neste caso, qualquer pessoa na posse do dispositivo poderá entrar na aplicação em estudo e, por sua vez, entrar na conta do indivíduo sem qualquer tipo de esforço.

Desta forma o grupo, o grupo sugere que, à semelhança do que acontece por exemplo numa aplicação de *smartphone* de um banco e da conta bancária, o Utilizador deverá sempre introduzir um código de Acesso para usufruir da sua conta. Este tipo de código de acesso poderá ser:

- Impressão Digital;
- PIN de 6 dígitos;
- Padrão de desbloqueio;
- Face ID.

Lembrando que esta recomendação não oferece proteção total da conta de um indivíduo, no entanto, irá dificultar muito o trabalho de quem tentar entrar na conta do mesmo sem permissão.

8.2 SQL Injection Protection

Dado que ainda agora falamos de Login, será então importante falar sobre a proteção do Sistema face a ataques denominadas como **SQL Injection**. Este, que é um dos ataques mais comuns da Internet, que os *attackers* utilizam para poder código SQL arbitrário numa dada aplicação, colocando o mesmo em campos de texto que poderão fazer parte da execução de um *statement*, como poderá ser o caso de uma password. Por esse motivo, também nos devemos defender contra este tipos de ataques. Para isso podemos utilizar algumas das seguintes técnicas:

- **Parameterized Statements** - parâmetros (ou seja, entradas) passados para instruções SQL tratados de maneira segura;

- **Object Relational Mapping;**
- **Escaping Inputs** - se não é possível usar Parameterized Statements, ou uma biblioteca que escreva SQL por nós, é garantir o *escape* adequado de caracteres de string especiais nos parâmetros de entrada;
- **Sanitizing Inputs.**

Mais uma vez, esta é uma recomendação que cobre apenas os ataques do tipo SQL Injection por isso ainda existem muitas mais otimizações de segurança que podem ser tomadas.

8.3 Assinatura de Mensagens e Encriptação (Portador - Verificador)

Como podemos ver pela maquete e enunciado fornecido, a comunicação entre o Portador e o Verificador é feita em vários tipos de tecnologias possíveis, às quais já apontamos vulnerabilidades anteriormente. Sabemos ainda que as mensagens e conteúdos são trocadas no formato CBOR - Concise Binary Object Representation.

Deste modo é necessário reforçar a robustez desta comunicação, dado que queremos alcançar e garantir o máximo de confiabilidade possível e todo o Sistema.

Para isso, a equipa acha por bem realizar a encriptação de todo o conteúdo trocado nesta comunicação, bem como estabelecer assinatura de mensagens para aumentar a segurança da mesma. Assim, o processo poderia ser o seguinte:

1. Gerar uma Assinatura no dispositivo;
2. Enviar chave pública para o outro dispositivo;
3. Receber chave pública do outro dispositivo;
4. Verificar assinatura da mensagem do outro dispositivo;
5. Gerar chave partilhada;
6. Criar Hash de chave partilhada e enviar Hash para o outro dispositivo;
7. Receber Hash de chave partilhada do outro dispositivo e verifica se é igual;
8. Criar mensagem cifrada com chave partilhada e enviar para o outro dispositivo;

Podemos perceber que estes passos dizem apenas respeito a um dos lados da comunicação. O outro dispositivo terá um funcionamento análogo, no entanto irá alterar a ordem com que recebe e envia as várias chaves e Hashs, bem como vai decifrar uma mensagem e não encriptar (como no caso dado acima).

Esta maneira de estabelecer a comunicação entre estes dois dispositivos (**Portador** e **Verificador**) torna a interação muito mais segura e robusta contra ataques de interceção da ligação, como são por exemplo, os ataques Man-In-The-Middle.

8.4 Sistema de Armazenamento

É também dito no Enunciado que, por vezes, quando o processo de verificação do documento mID é feito on-line, a entidade Emissora pode entrar em déficit de produtividade, e dependendo do número e tipo de serviços suportados, podemos criar uma carga significativa na mesma.

Por essa razão, a equipa pensa que à infra-estrutura já apresentada, devem ser acrescentados componentes que possam assegurar o melhor funcionamento da mesma. Para isso, achamos por bem adicionar uma quantidade generosa de **memória RAM**, capaz de suportar vários pedidos simultaneamente e ainda um vasta quantidade de **memória SSD** (preferencialmente, dado que os pedidos em memória em HDD são mais lentos do que em SSD), de forma a poder ter guardados todos os atributos necessários de todos os clientes do Serviço.

8.5 Proteção Binária (Rooting/Jailbreak)

Efetuar um **Root (Android)** ou um **Jailbreak (IOS)** de um dispositivo móvel contorna a proteção de dados e os esquemas criptográficos do sistema. Muitas vezes este tipo de operação é realizada quando os Utilizadores querem instalar aplicações e software diferente do que aquele que o **manufacturer** disponibilizou para esse dispositivo.

No entanto, quando um dispositivo é comprometido desta maneira, qualquer forma de código malicioso pode ser executado no dispositivo o que pode alterar significativamente os comportamentos pretendidos de uma aplicação.

Sendo assim, o melhor seria não deixar executar a nossa aplicação em dispositivos que foram alvo de Root ou Jailbreak. Para isso teria de ser implementada uma forma de deteção de Root/Jailbreak, que vai adicionar uma camada extra de proteção e mitigação de riscos da nossa aplicação, de forma a proteger os dados da mesma, para que estes não sejam indevidamente expostos.

9 Conclusão

Dado por concluído o último trabalho da Unidade Curricular de Tecnologias de Segurança, o grupo encontra-se realizado e feliz com o trabalho desenvolvido.

Ao longo do desenvolvimento deste projeto, tivemos algumas dificuldades a tentar perceber como a aplicação e o Sistema funcionavam, no entanto estas foram desvanecendo conforme nos informávamos mais sobre o assunto. Eventualmente, conseguimos implementar e estudar todas as técnicas que pretendemos, onde se enquadram: Catalogação de vulnerabilidades e exploits; Catalogação de fraquezas típicas; Modelação de ameaças; Análise de risco.

O projeto proposto pela equipa docente fez com que cada um de nós tivesse de ser obrigado a pensar fora da caixa, e a ver mais do que simples componentes de um Sistema. Fomos dessa maneira postos à prova, e acreditamos ter sucedido. Para além das aprendizagens a nível de *soft skills* aumentadas ao longo do desenvolvimento deste trabalho, consolidamos ainda ensinamentos das aulas que foram úteis e essenciais para a realização deste projeto.

Anexos

Anexo A - Matriz de Análise de Risco

		SEVERITY				RISK RATING KEY	
		Acceptable	Tolerable	Undesirable	Intolerable	LOW	
LIKELIHOOD	Improbable Risk is Unlikely to Occur	1 LOW	4 MEDIUM	7 MEDIUM	10 HIGH	MEDIUM	
	Possible Risk is Likely to Occur	2 LOW	5 MEDIUM	8 HIGH	11 EXTREME	HIGH	
	Probable Risk Will Occur	3 MEDIUM	6 HIGH	9 HIGH	12 EXTREME	EXTREME	

Figura 74: Matriz de Análise de Risco