**<Net Rogue>**

**Prasību analīze un idejas formulējums**

**Kursā Programmēšana**

**Versija 0.0.2**

|  |  |  |  |
| --- | --- | --- | --- |
| **Datums** | **Versija** | **Apraksts** | **Autors** |
| 2024.11.7 | 0.0.1 | Programmas idejas apraksts | Rihards Bukovskis |
| 2025.02.13 | 0.0.2 | Shēmas un to apraksts | Rihards Bukovskis |
|  |  |  |  |

**Darba autori:**

**Rihards Bukovskis**

**Darba termiņš:**

**27.11.2024. Ieskaitot**

**Saturs**

[1. Problēmas izpēte un analīze (15 punkti) 3](#_Toc181875993)

[2. Gitlab vai github repozitorijs – uzaicināts pasniedzējs (15 punkti) 4](#_Toc181875994)

[3. Darba noformēšana (5 punkti) 5](#_Toc181875995)

[Atsauces 6](#_Toc181875996)

# Problēmas izpēte un analīze (15 punkti)

Šis projekts būs dažādu WIFi ievainojamību apkopojums vienā rīkā. Lietotājam būs iespēja izvēlēties kādu no implementētajām ievainojamībām un to palaist, izmantojot savu tīkla karti. Tomēr programmai būs divi ierobežojumi. Tā var tikt palaista tikai uz linux sistēmas, jo uz windows ir daudz grūtāk darboties tīkla zemajā līmenī, kā arī lietotāja tīkla kartei ir jāatbalsta “Monitor Mode”. Ja tīkla kartei nav iespējams uzlikt Monitor Mode, tas nozīme, ka ar tīkla karti nav iespējams noklausīties visu satiksmi, kas to padara neizmantojamu WiFi uzbrukumiem.

Sākumā programmā būs iespējams palaist 3+1 ievainojamību:

1. Programmā tiks iekļauti OWE uzbrukumi, kas tika izstrādāti mana ZPD laikā. Tiks izveidoti visi 3 uzbrukumi un ir sagaidāms, ka, tos izstrādājot C valodā, tie būs efektīvāki nekā iegūts ZPD darbā.
2. Tiks iekļauts Twin AP uzbrukums, kurā tiek viltots piekļuves punkts. Šo uzbrukumu, visticamāk, ka būs sarežģītāk implementēt nekā citus, daudzo mainīgo dēļ. Piemēram, atkarībā no drošības protokola ir vairāki varianti, tad vēl ir piekļuves punkta specifiskie parametri, kā piemēram, datus pārraides ātrumi.
3. Papildus pie Twin AP būs iespēja uzlikt SSID Stripping, kur tiktu pievienoti papildus īpašās rakstzīmes, lai viltus piekļuves punkta SSID izskatītos tāpat, kā īstā.
4. Pēdējā ievainojamība, ko būs iespējams testēt ir ievainojamība, kur, izmantojot noklausīto PMKID no WPA2 tīkla ir iespējams uzlauzt tīkla paroli. Programma pievienosies pie piekļuves punkta līdz 4-pušu rokasspiediena pirmajai ziņai – tajā ir minēts PMKID. To izmantojot, ir iespējams lokāli iegūt paroli.

Šis projekts ir domāts, manas C valodas pielietojuma prasmju uzlabošanai un labākai WiFi ievainojamību izprašanai. Ir viegli lasīt par ievainojamībām un domāt, ka viss ir saprasts, tomēr reāli mēģinot tās implementēt, ir iespējams tās ļoti labi izprast, kā arī implementēšanas procesā var rasties arī jaunu ievainojamību idejas. Šis projekts ir arī domāts manu C valodas prasmju uzlabošanai. Iepriekš es esmu izstrādājis dažas ievainojamības Python programmēšanas valodā. Lai gan ir viegli uztaisīt ātrus ievainojamību testus, tie ne vienmēr izpilda nepieciešamās prasības, piemēram, tie nebūs pietiekami ātri, ņemot vērā, ka Python ir augsta līmeņa programmēšanas valoda. Tāpēc es esmu izvēlējies šo projektu izveidot, izmantojot C valodu, kas sniedz zema līmeņa piekļuvi tīklam un paketēm, kā arī C valodai ir “zema aprēķinu slodze” (no ang. Val. “Low computational overhead”), kas nozīmē, ka būs iespējams izveidot daudz ātrākas programmas, kas dažos gadījumos var uzlabot uzbrukuma izdošanās efektivitāti. Labs piemērs šim ir viens no maniem ZPD uzbrukumiem, kur tika viltota klienta asociācijas pieprasījuma pakete. Ar Python šim uzbrukumam bija tikai 10% efektivitāte, jo pakete bija jāizsūta ļoti īsā laika sprīdī. Mana teorija ir, ka, ja šo pašu uzbrukumu implementētu C valodā, tad uzbrukuma efektivitāte ievērojami uzlabotos. Kā arī ir daži uzbrukumi, kuriem ir nepieciešams ļoti liels ātrums, lai vispār tie izdotos, kā piemēram, “reactive jamming” uzbrukums, kur tikai konkrētas paketes tiek bloķētas. Tas notiek reāllaikā analizējot paketes, kas tiek sūtītas, un, ja konkrētā paketē atbilst kritērijiem, tad atlikusī paketes daļa tiek bloķēta. Tāpēc pēc iespējams implementējamo uzbrukumu analīzes, esmu izvēlējies šo projektu izstrādāt C valodā.

Šīs programmas galvenie lietotāji būtu WiFi entuziasti, kas interesējas par drošību. Ar šīs programmas palīdzību lietotāji būs spējīgi paši izmēģināt uzbrukumus, lai, piemēram, pārbaudītu vai viņu tīkla iekārtas nav ievainojamas pret dotajiem uzbrukumiem.

# Gitlab vai github repozitorijs – uzaicināts pasniedzējs (15 punkti)

Izveidots repozitorijs, piekļuve grupas biedriem un pasniedzējam (5p) Linku ielikt šeit dokumentā

<https://github.com/rihrds/NetRogue>

Ievietots readme fails ar tekstu no šī dokumenta par idejas aprakstu (īsā, kodolīgā versija) (5p)

Uzskaitīti plānotie projekta ierobežojumi (5p) – laika, atmiņas, ierīču (mobilās/datora vai tikai datora utml.) u.c.

Programma strādās tikai uz linux ierīcēm, jo linux piedāvā visērtāko pieeju zemajam tīkla līmenim, kamēr ar windows to izdarīt ir daudz sarežģītāk, un tikai ar tīkla kartēm, kuras ir iespējams ielikt Monitor mode, jo tikai tā var noklausīt visu tīkla satiksmi.

# Darba noformēšana (5 punkti)

..

# 4. Shēmas un to apraksti

A diagram of a program

Description automatically generated

Šajā diagrammā aprakstīts, kā notiks Twin AP uzbrukuma izpilde. No sākuma klients palaiž programmu un tai dod argumentu, ka vēlas izpildīt uzbrukumu. Klients var dot papildus argumentus, piemēr, konkrētu kanālu uz kura vajadzētu darboties viltus Ap, ar kādu SSID tam darboties.. Tālāk programma pārbauda, vai lokāli jau eksistē AP ar doto nosaukumu, ja tas eksistē, tad programma nokopē AP parametrus no tā beacon paketēm, lai būtu pēc iespējas līdzīgāks īstajam AP. Ja neatrod AP ar doto SSID, tad pārējos klienta nepiedāvātos parametrus tas izvēlas pats, piemēram, kanālu tas izvēlas visneaizņemtāko. Tālāk tas sāk pārraidīt savas Beacon paketes, mēģinot piesaistīt klientus, lai tie pievienotos viltus tīklam. Kad kāds lietotājs, mēģina pievienoties, programma to veiksmīgi asociē un paziņo lietotājam, ka ir kāds pievienojies pie viltus AP. Tālāk programma veic upura pakešu maršrutēšanu caur kādu tīklu, pie kura tas ir pievienojies, lai upuris tik un tā spētu piekļūt internetam, bet satiksmi būtu iespējams noklausīt.

A diagram of a program

Description automatically generated

Šis uzbrukums ir paņemts no mana ZPD darba, kur programma viltos Piekļuves punkta asociācijas atbildes paketi. Šis uzbrukums balstās uz principa, ka klients pirmo apstrādās viltoto asociācijas atbildes paketi un ignorēs īsto paketi.

Programma vispirms savienosies ar īsto piekļuves punktu, lai iegūtu pēc iespējas pareizāku asociācijas atbildi. Ja tas mēģinātu uztaisīt pats savu asociācijas atbildi, tad iespējams tas tiktu neprecīzi uztaisīts, kas nozīmē, ka klients to neapstrādātu un uzbrukums būtu neizdevies. Tātad programma no sākuma pievienosies pie īstā piekļuves punkta, iegūs tā asociācijas atbildi, to saglabās un atvienosies. Tālāk tas sāks klausīties, kad kāds klients sāks pievienoties pie īstā piekļuves punkta (šo programmu ir iespējams arī mērķēt uz konkrētiem klientiem, norādot to mac adresi). Kad tiks uztverts, ka piekļuves punkts izsūta autentifikācijas atbildi (atkarībā no programmas ātrumu, programma varētu arī reaģēt uz asociācijas pieprasījumu), programma izsūtīs viltoto asociācijas atbildi. Vienīgais veids, kā programma spēs noteikt, vai uzbrukums ir izdevies, ir, klausoties, vai netiek nosūtīta 4-pušu rokasspiediena 3. Ziņa. (Par veiksmes gadījumu vēl tiks domāts tālāk, jo kā ZPD rāda, dažām ierīcēm vajag 15 sekundes, lai padotos uz savienošanos ar piekļuves punktu)

A diagram of a program

Description automatically generated with medium confidence

Šajā uzbrukumā programma mēģinās pieslēgties pie visiem klienta minētajiem piekļuves punktiem, lai iegūtu to PMKID no to EAPOL ziņas, kas ir 4-pušu rokasspiediena 1. Ziņā atrodams. Klients var norādīt vienu, vairākus, vai visus lokālos AP, kuriem uzbrukt. Tālāk programma pēc kārtas, pievienosies pie šiem AP, ja tie eksistē un iegūs to PMKID. Pēc PMKID iegūšanas programma paziņos, ka ir veiksmīgi iegūts konkrētā piekļuves punkta PMKID, un tiks sākta PMKID uzlaušana ar Bruteforce metodi, kur tiks mēģinātas miljoniem iespējamo paroļu pēc kārtas, ja, izejot cauri visām populārākajām parolēm cauri, neizdodas uzlauzt PMKID, tad PMKID tiks aizsūtīts lietotājam ar ziņu, ka nav programmai neizdevās to uzlauzt.

Šis uzbrukums balstās uz hashcat galvenā izstrādātāja atklājuma 2016. Gadā. Šī metode ir labāka par iepriekš plaši izmantoto metodi, kur, lai varētu mēģināt uzlauzt tīkla paroli bija jānoklausa pareizi visas 4-pušu rokasspiediena ziņas. Šim uzbrukumam vajag noklausīt tikai 1. Ziņu. Šis uzbrukums nav tik efektīvs pret WPA3 tīkliem, jo to šifrēšanas metodes ir advancētākas, tomēr WPA2 vēl ar vien tiek plaši izmantoti.

A diagram of a program

Description automatically generated

# Piemērotākie rīki un bibliotēkas

Kā jau idejas konceptā tika minēts, tad programmas izveidei es izmantošu C valodu, tās zemā līmeņa un ātrumu dēļ.

Tīkla ziņu uztveršanai un izsūtīšanai ir precīzaas bibliotēkas vēl tiek meklētas, tomēr pagaidām galvenais kandidāts ir libpcap bibliotēka, kas piedāvā ērtu veidu, kā analizēt tīkla satiksmi. Šo pašu bibliotēku izmanto arī Wireshark un tcpdump. Ar šo bibliotēku ir iespējams reāllaikā notvert paketes, un tās saglabāt, bet iespēja izsūtīt paketes ar šo bibliotēku ir diezgan sarežģīti.

Tomēr alternatīva libpcap ir nl80211 bibliotēka, kas piedāvā tās pašas iespējas, ko libpcap, kā arī pakešu nosūtīšanu, tomēr ar samazinātu lietošanas ērtumu.

Tiks analizētas abu bibliotēku iespējas un attiecīgi tiks izvēlēta piemērotākā šai programmai.

# Atsauces

1. <https://aireye.tech/2021/09/13/the-ssid-stripping-vulnerability-when-you-dont-see-what-you-get/>
2. <https://hashcat.net/forum/thread-7717.html>
3. ....