

Detekcia nepovoleného prístupu kustomizáciou nízko interaktívnych honeypotov

Príprava nástrojov pre kustomizovateľné honeypoty

Jakub Perdek

Rozdelenie honeypotov

Honeypoty sú bezpečnostným zdrojom generujúcim upozornenia pri zachytení interakcie s nimi. Napríklad v podobe ich prieskumu, útoku alebo pri ich kompromitácii.

Podľa ich nasadenia:

výskumné

- Obvykle zvonka dostupné
- Získavanie nových poznatkov o spôsobe útoku

produkčné

- Obvykle v intranete
- Zistenie nepovoleného prístupu v intranete

Podľa stupňa interaktivity:

Kritérií návrhu

Nízko interaktívne

- Nízka interaktívnosť
- Menej informácií o útočníkovi
- Napr. honeytokens, Stug

Stredne interaktívne

- Prepracovanejšie simulácie prostredia
- Napr. Cowrie

Vysoko interaktívne

- Vysoká interakcia s útočníkom
- Vyššie riziko zneužitia

Detekcia nepovoleného prístupu (honeypot / honeypot)

Analyzované / navrhnuté prístupy

- ▶ Tvorba súboru pri ktorom sa zabezpečí zaznamenanie aktivity prístupu k nemu
- ▶ Simulácia protokolu SSH alebo Telnet
 - ▶ Možnosť odhaliť získané prístupové údaje
 - ▶ V rámci vnútornej simulácie možno odhaliť aj útočnickové aktivity
- ▶ Klonovanie existujúcej funkcionality s pridaním detekčnej funkcionality
- ▶ Tvorba fiktívneho odkazu v konfiguračnom súbore služby
 - ▶ Napr. služby poskytujúcej odkazy na jednotlivé stránky webového letáku

Návrh nízko interaktívneho honeytokenu

Ako zabezpečiť detekciu prístupu k honeytokenu?



Detekcia na úrovni súborového systému (príkazy pre jeho otvorenie,...)



Detekcia zaznamenaním informácie (o vstupe,...) pri prístupe k súboru

Vlastnosť
honeypotu:

K honeytokenu by nikto nemal pristupovať (okrem narušiteľa)



Každý prístup k nim môže byť potenciálnym narušením systému

(Môže to byť aj false pozitívum, ktorých je ale veľmi málo)

Dôležitý
predpoklad pre
realizáciu:

Dôležité je zaznamenať prístup k informácii samotnej



Samotné otvorenie súboru nemusí byť vždy detegované

MOŽNO REALIZOVAŤ V RÁMCI HONEYTOKENU SAMOTNÉHO!

Zaznamenanie nepovoleného prístupu k webovému honeytokenu

Zápis informácie do súboru

Bez servera nie je umožnený zápis ďalších súborov v súborovom systéme klienta

Odoslanie informácie na server / databázy

Odoslanie informácií o aktivite aj s bežným obsahom

Zápis do dočasnej pamäti

Problematické zaručiť perzistenciu a bezpečné doručenie informácie

Určené predpoklady:

- ➔ POTREBA ZABEZPEČIŤ UTAJENIE INFORMÁCIE
- ➔ POTREBA ZABEZPEČIŤ PODMIENENIE ODOSLANIA LOGU VÝMENOU ZA HĽADANÚ / KLÚČOVÚ INFORMÁCIU

Prístupy pre zaznamenanie nepovoleného prístupu k obsahu

Použitie iframe elementu pre načítanie dodatočného obsahu

Zaznamenanie požiadavky o tento obsah

Vloženie kódu priamo odosielajúceho informácie o aktivite na server

Zaznamenanie požiadavky o tento obsah



POTREBA ZABEZPEČIŤ UTAJENIE INFORMÁCIE

- HASHOVANIE OBSAHU
- UKRYTIE FUNKCIONALITY VO FIKTÍVNEJ BIZNIS LOGIKE



POTREBA ZABEZPEČIŤ PODMIENENIE ODOSLANIA LOGU VÝMENOU ZA HĽADANÚ / KLÚČOVÚ INFORMÁCIU

- ZAMEDZIŤ ODCHYTENIU LOGOVACEJ INFORMÁCIE VÝMENOU ZA OBSAH (alebo kľúču k nemu)

Tvorba nízko interaktívneho honeypotu

Klonovanie
tokenov z danej
domény

- HTTrack
- pywebcopy

Injektnutie
iframe elementu
s príslušným
odkazom na
obsah

- Vloženie url s
vlastným obsahom
(na ďalší webový
dokument) alebo
url na proxy server

Použitie iframe elementu pre
načítanie dodatočného obsahu

Minifikácia
webových
dokumentov

- Minifikácia:
- CSS (Yui Compressor)
- Javascriptu (Google Closure Compiler)
- HTML (htmlmin)

Finalizácia
kustomizácie

- Nastavenie
parametrov
- Úprava HTML

Tvorba nízko interaktívneho honeypotu

Klonovanie tokenov
z danej domény

- HTTrack
- pywebcopy

Tvorba detekčného
skriptu a
prispôbenie
serveru

- Stub v javascripte pre
• honeytoken a server

Hashovanie a
zneprístupňovanie
obsahu bez kľúča k
nemu

- Gzip
- Deflate
- Brotli
- Base64, Base32, Base85
- Url nástroje
- Rozdelenie a skladanie
kódu

Minifikácia
webových
dokumentov

- Minifikácia:
- CSS (Yui Compressor)
- Javascriptu (Google
Closure Compiler)
- HTML (htmlmin)

Finalizácia
kustomizácie

- Nastavenie
parametrov
- Úprava HTML

Vloženie kódu priamo odosielajúceho
informácie o aktivite na server

Kopírovanie webovej lokality

Použitie vlastného skriptu s nástrojom HTTrack:

```
"/venv/Scripts/python.exe" ./content_downloader/website_copier_wrapper.py -  
-folder ./download --url https://www.bestoldgames.net/battle-chess  
-vkladá text o kopírovaní do jednotlivých súborov
```

Použitie vlastného skriptu s knižnicou pywebcopy:

```
"/venv/Scripts/python.exe" ./content_downloader/page_parser.py --url  
https://www.bestoldgames.net/battle-chess  
-zle pracuje s relatívnymi adresami - najnovšia verzia  
-nemá možnosť jednovláknového spracovania - stable verzia  
Zasekáva sa pri kopírovaní súborov
```

Minifikácia jednotlivých súborov

CSS / Javascript / HTML

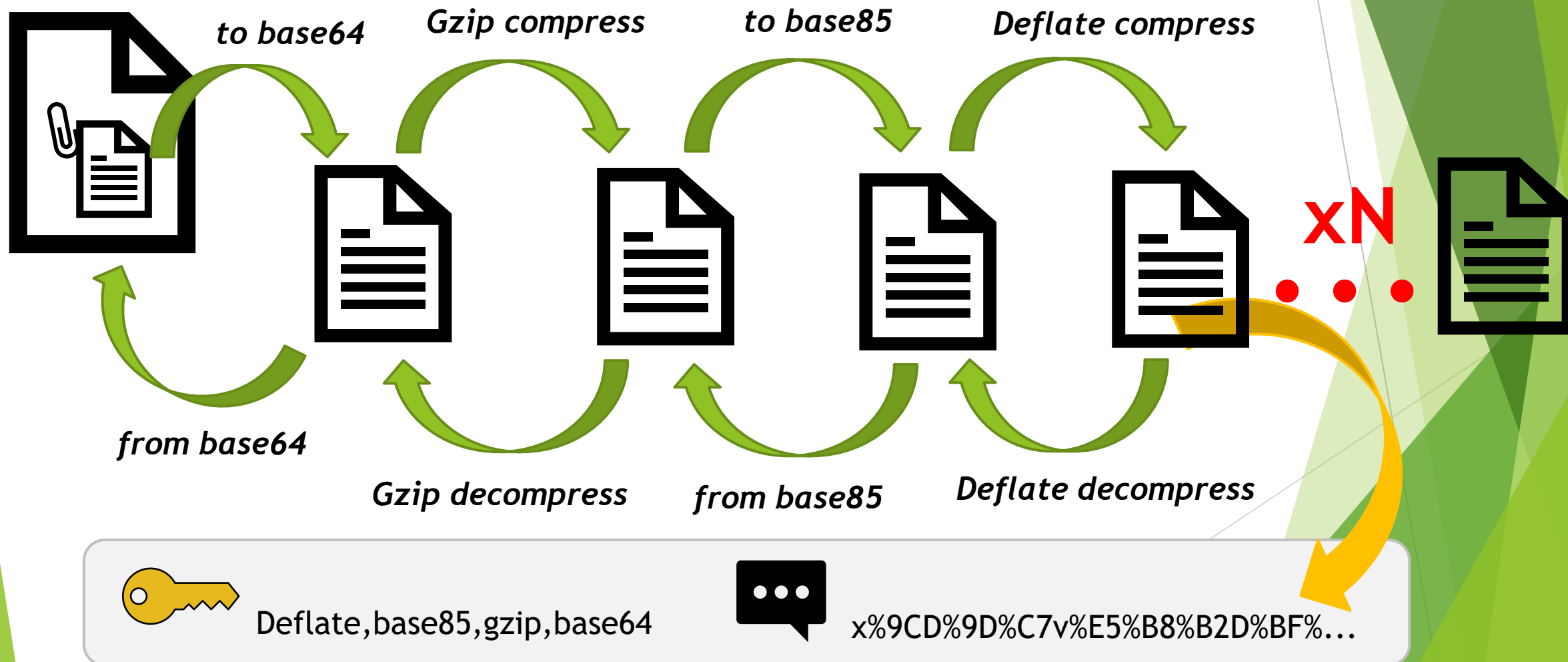
- ▶ Tvorba vlastného skriptu integrujúceho minifikačné nástroje
 - ▶ Spracovanie celého stromu dokumentov
 - ▶ Zamlženie obsahu
 - ▶ Zmenšenie veľkosti súborov

TESTOVANIE MINIFIKÁCIE STROMU DOKUMENTOV:

```
"./venv/Scripts/python.exe" ./content_minifier/content_minifier.py ./download
```

Ukrytie obsahu dokumentu

- Implementácia integrácie nástrojov ukrývajúci alebo komprimujúci konkrétny obsah



Vytvorená funkcionálna pre ukrytie obsahu

- ▶ Tvorba hashu z reťazca
- ▶ Komprimácia obsahu pomocou niektorej z metód
 - ▶ Prevod výslednej binárnej reprezentácie na reťazec - quotovanie
- ▶ Úprava reťazcov pomocou metód pre enkódovanie / dekódovanie url
- ▶ Náhodné aplikovanie metód pri ukrývaní obsahu
- ▶ Zabezpečenie spätnej konverzie na pôvodný reťazec
- ▶ Umožnenie aplikovať konverziu zvolený počet krát
- ▶ Možnosť vybrať si množinu metód, ktoré sa majú použiť




TESTOVANIE FUNKČNOSTI:

```
"./venv/Scripts/python.exe" ./content_concealing/content_concealing.py
```

Injektovanie obsahu do Iframe

- ▶ Tvorba servera pre logovanie aktivity v honeytokenu
- ▶ Injektovanie iframu do honeytokenu s odkazom na stránku servera, ktorá bude aktivity logovať

POUŽITIE:

- ▶ `"./venv/Scripts/python.exe" ./tracking_injector/inject_iframe.py`
 - ▶ `--html_file ./download/www.bestoldgames.net/battle-chess.html`  Cesta k honeytokenu
 - ▶ `--tracking_address http://localhost:5001/render.html`  Adresa, ktorá sa dosadí do atribútu src konkrétneho / vytvoreného iframu
 - ▶ `--content_folder ./content_folder`  Adresa, kde sa majú vložiť súbory z predchádzajúceho obsahu iframu pokiaľ bol

Inštalácia nízko interkatívneho honeypotu Thug

Detekcia exploitu:

Testovanie schopnosti detekcie exploitu:

```
if __name__ == "__main__":  
    t = TestAPI()  
    t.analyze_local("./thug/tests/samples/exploits/2448.html")  
    #t.analyze("http://www.google.com")
```

```
ubuntu@ubuntu1804:~/Desktop/thug$ ./bin/python3 test.py
```

```
[2022-03-18 16:59:05] ActiveXObject: microsoft.xmlhttp  
[2022-03-18 16:59:06] ActiveXObject: webviewfoldericon.webviewfoldericon.1  
[2022-03-18 16:59:06] [WebViewFolderIcon ActiveX] setSlice(2147483646, 84215045,  
84215045, 84215045)  
[2022-03-18 16:59:06] [WebViewFolderIcon ActiveX] setSlice attack  
[2022-03-18 16:59:06] [EXPLOIT Classifier] URL: ./thug/tests/samples/exploits/24  
48.html (Rule: CVE-2006-3730, Classification: )  
[2022-03-18 16:59:06] ActiveXObject: webviewfolderi48.html (Rule: CVE-2006-3730, Classification: )  
[2022-03-18 16:59:06] [WebViewFolderIcon ActiveX] s[2022-03-18 16:59:06] [LIBEMU][Shellcode Profile] UINT WINAPI WinExec (  
84215045, 84215045) LPCSTR lpCmdLine = 0x417085 =>  
[2022-03-18 16:59:06] [WebViewFolderIcon ActiveX] s = "calc";  
UINT uCmdShow = 0;  
) = 0x20;  
LPTOP_LEVEL_EXCEPTION_FILTER SetUnhandledExceptionFilter (  
LPTOP_LEVEL_EXCEPTION_FILTER lpTopLevelExceptionFilter = 0x0 =>  
none;  
) = 0x7c81cdda;  
  
[2022-03-18 16:59:06] Thug analysis logs saved at ../logs/44b15fa4daad4e2c2b0c97  
5d05014350/20220318165902  
ubuntu@ubuntu1804:~/Desktop/thug$
```

Skript pre lokálnu analýzu súborov s použitím Thug API

```
from thug.ThugAPI import ThugAPI

class TestAPI(ThugAPI):
    def __init__(self):
        ThugAPI.__init__(self)

    def analyze_local(self, url):
        # Set useragent to Internet Explorer 9.0 (Windows 7)
        self.set_useragent('win7ie90')

        # Set referer to http://www.honeynet.org
        self.set_referer('http://www.honeynet.org')

        # Enable file logging mode
        self.set_file_logging()

        # Enable JSON logging mode (requires file logging mode enabled)
        self.set_json_logging()

        # [IMPORTANT] The following three steps should be implemented (in the exact
        # order of this example) almost in every situation when you are going to
        # analyze a remote site.

        # Initialize logging
        self.log_init(url)

        # Run analysis
        self.run_local(url)

        # Log analysis results
        self.log_event()
```

Použitá literatura

- ▶ BERCOVITCH, Maya, Meir RENFORD, Lior HASSON, Asaf SHABTAI, Lior ROKACH a Yuval ELOVICI, 2011. HoneyGen: An automated honeytokens generator. V: 2011 IEEE International Conference on Intelligence and Security Informatics (ISI 2011): Proceedings of 2011 IEEE International Conference on Intelligence and Security Informatics [online]. Beijing, China: IEEE, s. 131-136 [cit. 25.2.2022]. ISBN 978-1-4577-0082-8. Dostupné na: doi:10.1109/ISI.2011.5984063
- ▶ CABRAL, Warren, Craig VALLI, Leslie SIKOS a Samuel WAKELING, 2019. Review and Analysis of Cowrie Artefacts and Their Potential to be Used Deceptively. V: 2019 International Conference on Computational Science and Computational Intelligence (CSCI): 2019 International Conference on Computational Science and Computational Intelligence (CSCI) [online]. Las Vegas, NV, USA: IEEE, s. 166-171 [cit. 21.2.2022]. ISBN 978-1-72815-584-5. Dostupné na: doi:10.1109/CSCI49370.2019.00035
- ▶ MANSOORI, Masood, Ian WELCH a Qiang FU, 2014. YALIH, Yet Another Low Interaction Honeyclient. New Zealand. 2014, roč. 149, s. 9.
- ▶ MOHAMMED, Mohssen a Habib-ur REHMAN, 2015. Honeypots and Routers: Collecting Internet Attacks [online]. 0. vyd. B.m.: Auerbach Publications [cit. 21.2.2022]. ISBN 978-0-429-17195-6. Dostupné na: doi:10.1201/b19660
- ▶ MOORE, Chris a Ameer AL-NEMRAT, 2015. An Analysis of Honeypot Programs and the Attack Data Collected. V: Hamid JAHANKHANI, Alex CARLILE, Babak AKHGAR, Amie TAAL, Ali G. HESSAMI a Amin HOSSEINIAN-FAR, ed. Global Security, Safety and Sustainability: Tomorrow's Challenges of Cyber Security [online]. Cham: Springer International Publishing, Communications in Computer and Information Science, s. 228-238 [cit. 21.2.2022]. ISBN 978-3-319-23275-1. Dostupné na: doi:10.1007/978-3-319-23276-8_20

Použitá literatura

- ▶ NG, Chee Keong, Lei PAN a Yang XIANG, 2018. Honeypot Frameworks and Their Applications: A New Framework [online]. Singapore: Springer Singapore. SpringerBriefs on Cyber Security Systems and Networks [cit. 21.2.2022]. ISBN 978-981-10-7738-8. Dostupné na: doi:10.1007/978-981-10-7739-5
- ▶ RETI, Daniel a Norman BECKER, 2021. Escape the Fake: Introducing Simulated Container-Escapes for Honeypots. arXiv:2104.03651 [cs] [online]. 2021 [cit. 21.2.2022]. Dostupné na: <http://arxiv.org/abs/2104.03651>
- ▶ SANDERS, Chris, 2020. Intrusion detection honeypots: detection through deception. ISBN 978-1-73518-830-0.
- ▶ SMOKESCREEN TEAM, Open Source Honeypots That Detect Threats For Free [online]. Dostupné na: <https://www.smokescreen.io/practical-honeypots-a-list-of-open-source-deception-tools-that-detect-threats-for-free/>
- ▶ SPITZNER, L., 2002. Honeypots: Tracking Hackers. USA: Addison-Wesley Longman Publishing Co., Inc. ISBN 0-321-10895-7.
- ▶ ZULKURNAIN, Nurul Fariza, Azli Fitri REBITANIM a Noreha Abdul MALIK, 2018. Analysis of THUG: A Low-Interaction Client Honeypot to Identify Malicious Websites and Malwares. V: 2018 7th International Conference on Computer and Communication Engineering (ICCCE): 2018 7th International Conference on Computer and Communication Engineering (ICCCE) [online]. Kuala Lumpur: IEEE, s. 135-140 [cit. 21.2.2022]. ISBN 978-1-5386-6992-1. Dostupné na: doi:10.1109/ICCCE.2018.8539257

Automatizácia vloženia detegujúceho skriptu to webového tokenu

1. Predpripravenie skriptu s kódom, ktorý odosiela hlásenia na server
2. Tvorba kópie daného webového dokumentu / lokality
3. Načítanie detegujúceho skriptu
4. Zamaskovanie obsahu metódami pre skrytie obsahu
5. Rozdelenie obsahu do niekoľkých riadkov kódu, v ktorých sa bude zlučovať a umiestnenie ich do elementu skript
6. Zavolanie metódy eval() pre aplikovanie výsledného vytvoreného reťazca na konci tohto skriptu
7. Vloženie tohto script elementu do výsledného dokumentu
8. Pripravenie API serveru pre zachytenie hlásení

TESTOVANIE FUNKČNOSTI (zatiaľ bez argumentov):

```
"./venv/Scripts/python.exe" ./honey_token_generator/honey_token_constructor.py
```

Odkaz na repozitár

► [**https://github.com/jperdek/tokenCreator**](https://github.com/jperdek/tokenCreator)

Vyžadajte si prístup...