SPECIAL REPORT

FireEye

OPERATION SAFFRON ROSE

2013

Authors: Nart Villeneuve, Ned Moran, Thoufique Haq and Mike Scott

SECURITY REIMAGINED

CONTENTS

We believe we're seeing an evolution and development in Iranian-based cyber activity. In years past, Iranian actors primarily committed politically-motivated website defacement and DDoS attacks.[1] More recently, however, suspected Iranian actors have destroyed data on thousands of computers with the Shamoon virus,[2] and they have penetrated the Navy Marine Corps Intranet (NMCI), which is used by the U.S. Navy worldwide.[3]

In this report, we document the activities of the Ajax Security Team, a hacking group believed to be operating from Iran. Members of this group have accounts on popular Iranian hacker forums such as ashiyane[.]org and shabgard[.]org, and they have engaged in website defacements under the group name "AjaxTM" since 2010. By 2014, the Ajax Security Team had transitioned from performing defacements (their last defacement was in December 2013) to malware-based espionage, using a methodology consistent with other advanced persistent threat actors in this region.

It is unclear if the Ajax Security Team operates in isolation or if they are a part of a larger coordinated effort. The Ajax Security Team itself uses malware tools that do not appear to be publicly available. We have seen this group leverage varied social engineering tactics as a means to lure their targets into infecting themselves with malware. Although we have not observed the use of exploits as a means to infect victims, members of the Ajax Security Team have previously used publicly available exploit code in web site defacement operations.

In sum, FireEye has recently observed the Ajax Security Team conducting multiple cyber espionage operations against companies in the defense industrial base (DIB) within the Unites States, as well as targeting local Iranian users of anti-censorship technologies that bypass Iran's Internet filtering system.

Background

The transition from patriotic hacking to cyber espionage is not an uncommon phenomenon. It typically follows an increasing politicization within the hacking community, particularly around geopolitical events. This is followed by increasing links between the hacking community and the state, particularly military and/or intelligence organizations.

In the late 1990's and early 2000's, a similar transition occurred within the Chinese hacking community. During that time period, the Chinese hacking community engaged in website defacements and denial of service attacks in conjunction with incidents such as the accidental bombing of the Chinese embassy in Belgrade in 1999, the collision of a U.S. spy plane and a Chinese military plane in 2001, and the Japanese Prime Minister's controversial visit to the Yasukuni shrine in 2005.[4] Around this time a significant shift in philosophy began to take place.

Members of the Chinese hacking community that participated in such attacks soon found that transitioning to cyber espionage was more rewarding—both in terms of developing a more advanced skill set as well as in monetary remuneration. One group known as NCPH (Network Crack Program Hacker), whose founding member "Wicked/Withered Rose" was a patriotic hacker, made the transition to cyber espionage by founding a "hacker-for-hire" group

1 HP Security Research. "Threat Intelligence Briefing Episode 11". February 2014.

2 Perlroth, N. "In Cyberattack on Saudi Firm, U.S. Sees Iran Firing Back". October 2012.

3 Gallagher, S. "Iranians hacked Navy network for four months? Not a surprise". February 2014.

4 Key. "Honker Union of China to launch network attacks against Japan is a rumor". September 2010.

that simultaneously developed an association with the Chinese military.[5] The group began developing zero-day exploits, rootkits and remote access tools (RATs)—using them in attacks against a variety of targets including the U.S. Department of Defense.[6] (One of this group's associates, "whg", is still active and is believed to have developed one variant of the PlugX/SOGU malware.[7]) The rationale behind this transition within the Chinese hacking community is nicely summed up in a message by the "Honker Union of China" to its members in 2010:

What benefit can hacking a Web page bring our country and the people? It is only a form of emotional catharsis, please do not launch any pointless attacks, the real attack is to fatally damage their network or gain access to their sensitive information.[8]

In Iran, the hacking community appears to be undergoing a similar transformation. While a variety of Iranian hacker groups had engaged in politically motivated website defacements, the emergence of the "Iranian Cyber Army" in 2009 demonstrated "a concentrated effort to promote the Iranian government's political narrative online".[9] They targeted, among others, news organizations, opposition websites and social media.[10] This marked the beginning of a large-scale cyber offensive against the perceived enemies of the Iranian government.

Foreign news and opposition websites are routinely blocked in Iran, as are the tools that allow users in Iran to bypass these restrictions.[11] One of the key stakeholders in Iran's Internet censorship program is the Iranian Revolutionary Guard Corps (IRGC), under which the Basij paramilitary organization operates.

The Basij formed the Basij Cyber Council and actively recruits hackers in order to develop both defensive and offensive cyber capabilities.[12] There is increasing evidence to suggest that the hacker community in Iran is engaged in a transition from politically motivated defacements and denial of service attacks to cyber espionage activities. This model is consistent with the Basij's recruitment of paramilitary volunteer hackers to "engage in less complex hacking or infiltration operations" leaving the more technical operations to entities over which they have increasingly direct control.[13]

As such, the capabilities of threat actors operating from Iran have traditionally been considered limited.[14] However, the "Shamoon" attacks, which wiped computers in Saudi Arabia and Qatar, indicate an improvement in capabilities.[15] And unsurprisingly, Iran has reportedly increased its efforts to improve offensive capabilities after being targeted by Stuxnet and Flame.[16]

5 Elegant, S. "Enemies at The Firewall". December 2007. Dunham, K. & Melnick, J. "'Wicked Rose' and the NCPH Hacking Group". Wikipedia. "Network Crack Program Hacker Group".

6 Dunham, K. & Melnick, J. "'Wicked Rose' and the NCPH Hacking Group".

7 Blasco, J. "The connection between the Plugx Chinese gang and the latest Internet Explorer Zeroday". September 2012.

8 Key. "Honker Union of China to launch network attacks against Japan is a rumor". September 2010.

9 OpenNet Initiative. "After the Green Movement: Internet Controls in Iran 2009 – 2012". February 2013.

10 Rezvaniyeh, F. "Pulling the Strings of the Net: Iran's Cyber Army". February 2010. "Twitter hackers appear to be Shiite group". December 2009.

11 OpenNet Initiative. "Iran". June 2009.

12 The IRGC has also indicated that they would welcome hackers that support the Iranian government. Esfandiari, G. "Iran Says It Welcomes Hackers Who Work For Islamic Republic". March 2011, HP Security Research. "Threat Intelligence Briefing Episode 11". February 2014.

13 BBC Persian. "Structure of Iran's Cyber Warfare".

14 Mandiant. "M-Trends: Beyond the Breach, 2014", page 9. April 2014.

15 Mount, M. "U.S. Officials believe Iran behind recent cyber attacks". October 2012.

16 Shalal-Esa, A. "Iran strengthened cyber capabilities after Stuxnet: U.S. general". January 2013, Lim, K. "Iran's cyber posture". November 2013.

Attack Vectors

We have observed the Ajax Security Team use a variety of vectors to lure targets into installing malicious software and/or revealing login credentials. These attack vectors include sending email, private messages via social media, fake login pages, and the propagation of anti-censorship software that has been infected with malware.

Spear phishing

During our investigation, we discovered that these attackers sent targeted emails, as well as private messages through social media. For example, the attackers targeted companies in the DIB using a fake conference page as a lure to trick targets into installing malicious software. The attackers registered the domain "aeroconf2014[.]org" in order to impersonate the IEEE Aerospace conference—the conference's actual domain is aeroconf.org—and sent out an email with the following information:

From: invite@aeroconf2014[.]org
Subject: IEEE Aerospace Conference 2014

The email encouraged users to visit a fake conference website owned by the attackers:

Upon visiting the website, visitors were notified that they must install "proxy" software in order to access it, which is actually malware.

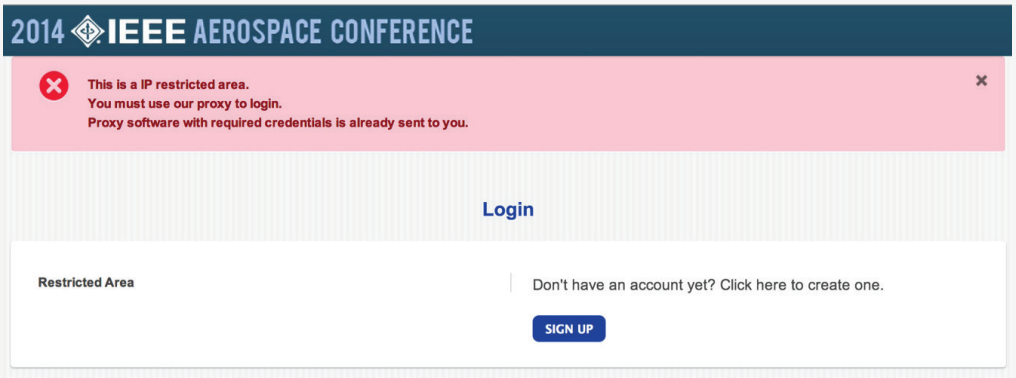


Figure 1: The Fake IEEE Aerospace Conference Website

[7] Bloomberg. "Neiman Marcus Hackers Set Off 60,000 Alerts While Bagging Credit Card Data." February 2014.

Credential Phishing

The attackers have also used phishing attacks, in which they set up Web pages to emulate various services that require security credentials. The attackers tailored these login pages for specific targets in the DIB and spoofed a variety of services such as Outlook Web Access and VPN login pages.

If users attempt to login through these fake Web pages, the attackers collect their login credentials.

Anti-censorship Tools

All Internet Service Providers (ISPs) in Iran are required to implement filtering technology that censors access to content which the Iranian government deems unacceptable.[17] This content includes categories such as pornography and political opposition.[18] In response to these restrictions, Iranians have been increasingly using software that bypasses such filtering technology.

To counter anti-censorship efforts, Iran has attempted to block the use of certain software tools.[19] In 2012, researchers found that an anti-censorship tool that is primarily used by Internet users in Iran was bundled with malware and redistributed.[20]

Our investigation found that malware-laden versions of legitimate anti-censorship software, such as Psiphon and Ultrasurf, were distributed to users Iran and Persian speaking people around the world.

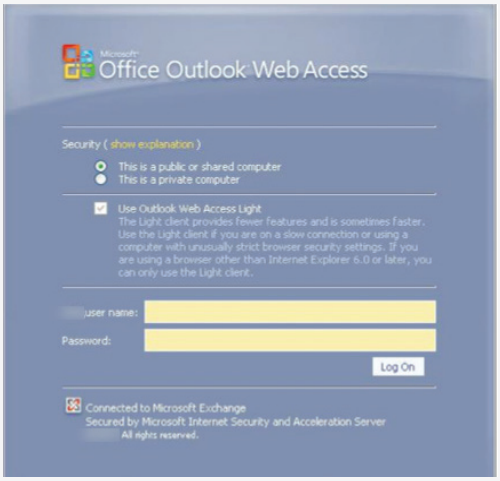


Figure 2: The Fake Outlook Web Access page

17 OpenNet Initiative. "Iran". June 2009.

18 OpenNet Initiative. "After the Green Movement: Internet Controls in Iran 2009 – 2012". February 2013.

19 Torbati, Y. "Iran blocks use of tool to get around Internet filter". March 2013.

20 Marquis-Boire, M. "Iranian anti-censorship software 'Simurgh' circulated with malicious backdoor". May 2012.

The "Stealer" Malware

Host-based Indicators and Malware Functionality

We have observed the Ajax Security Team use a malware family that they identify simply as 'Stealer'. They deliver this malware as a malicious executable (dropper). The executable is a CAB extractor that drops the implant IntelRS.exe. This implant, in turn, drops various other components into C:\Documents and Settings\{USER}\Application Data\IntelRapidStart\. The following files are written to disk in this location:

The IntelRS.exe is written in .NET and is aptly named "Stealer", as it has various data collection modules. It drops and launches AppTransferWiz.dll via the following command:

"C:\WINDOWS\system32\rundll32.exe" "C:\Documents and Settings\{USER}\Application

Data\IntelRapidStart\AppTransferWiz.dll",#110

110 is an ordinal that corresponds to "StartBypass" export in AppTransferWiz.dll.

File	Functionality
IntelRS.exe	Various stealer components and encryption implementation
DelphiNative.dll	Browser URL extraction, IE Accounts, RDP accounts (Imported by IntelRS.exe)
IntelRS.exe.config	Config containing supported .NET versions for IntelRS.exe
AppTransferWiz.dll	FTP exfiltration (Launched by IntelRS.exe)
RapidStartTech.stl	Base64 encoded config block containing FTP credentials, implant name, decoy name, screenshot interval and booleans for startup, keylogger and screenshot

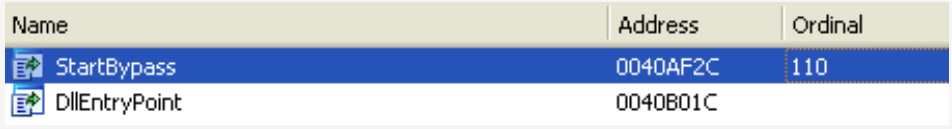


Figure 3: StartBypass Ordinal

Data exfiltration is conducted over FTP by AppTransferWiz.dll, which acts as an FTP client. This DLL is written in Delphi. There is code to exfiltrate data over HTTP POST as well, but it is unused. We also found incomplete code that would perform SFTP and SMTP exfiltration, which could be completed in a future version.

State is maintained between the stealer component IntelRS.exe and the FTP component AppTransferWiz.DLL using a file from the FTP server "sqlite3.dll", as well as a global atom "SQLiteFinish". IntelRS.exe waits in an indefinite loop, until AppTransferWiz.DLL defines this state.

Once the state is set, IntelRS.exe proceeds to collect data from various areas in the system as described below:

- Collects system information: hostname, username, timezone, IP addresses, open ports, installed applications, running processes, etc.
- Performs key logging
- Takes various screenshots
- Harvests instant messaging (IM) account information: GTalk, Pidgin, Yahoo, Skype
- Tracks credentials, bookmarks and history from major browsers: Chrome, Firefox, Opera
- Collects email account information
- Extracts installed proxy software configuration information
- Harvests data from installed cookies

IntelRS.exe loads a Delphi component called DelphiNative.DLL, which implements some additional data theft functionality for the following:

- Internet Explorer (IE) accounts
- Remote Desktop Protocol (RDP) accounts
- Browser URLs

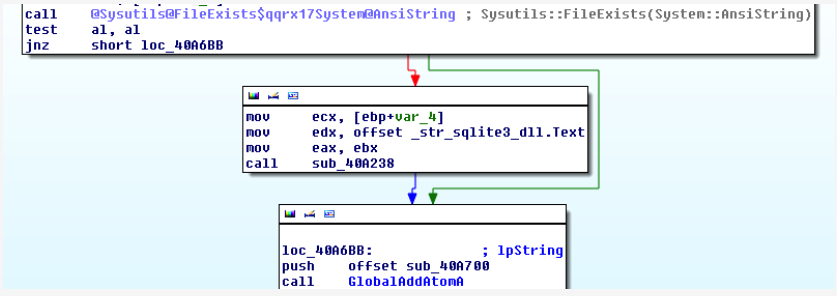


Figure 4: AppTransferWizard.dll creates sqlite3.dll and global atom

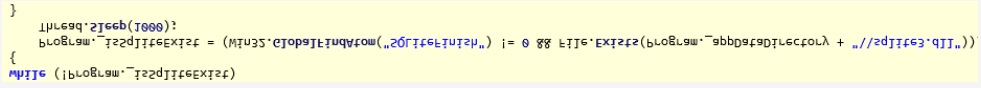


Figure 5: IntelRS.exe sleeps until global atom is set and sqlite3.dll is present

The Stealer component uses common techniques to acquire credential data. For instance, it loads vaultcli.DLL and uses various APIs shown below to acquire RDP accounts from the Windows vault.

Harvested data is encrypted and written to disk on the local host. The filenames for these encrypted files follow this naming scheme:

{stolen data type}_{victim system name}_ YYYYMMDD_HHMM.Enc

The {stolen data type} parameter indicates where the data was harvested from (e.g., a Web browser, an instant messenger application, installed proxy software).

Analysis of the malware indicates that the data is encrypted via a Rijndael cipher implementation; more specifically it uses AES which is a specific set of configurations of Rijndael. It uses a key size of 256 bytes and block size of 128 bytes, which conforms to the FIPS-197 specification of AES-256.[21] It utilizes the passphrase 'HavijeBaba' and a salt of 'salam!*%#' as an input to PBKDF2 (Password-Based Key Derivation Function 2) to derive the key and initialization vector for the encryption.[22] This key derivation implementation in .NET is done using the Rfc2898DeriveBytes class.[23] The passphrase and salt are Persian language words. "Havij" means "carrot", "Baba" means "father", and "Salam" is a common greeting that means "Peace".

Figure 6: Acquiring RDP Accounts

```
CODE:0040916D loc_40916D:                              ; CODE XREF: GetRDPAccounts+55↑j
CODE:0040916D                 push    offset aVaultenumerate ; "VaultEnumerateVaults"
CODE:00409172                 push    ebx             ; hModule
CODE:00409173                 call    GetProcAddress_0
CODE:00409178                 mov     [ebp+var_8], eax
CODE:0040917B                 push    offset aVaultopenvault ; "VaultOpenVault"
CODE:00409180                 push    ebx             ; hModule
CODE:00409181                 call    GetProcAddress_0
CODE:00409186                 mov     [ebp+var_C], eax
CODE:00409189                 push    offset aVaultclosevaul ; "VaultCloseVault"
CODE:0040918E                 push    ebx             ; hModule
CODE:0040918F                 call    GetProcAddress_0
CODE:00409194                 mov     [ebp+var_10], eax
CODE:00409197                 push    offset aVaultenumera_0 ; "VaultEnumerateItems"
CODE:0040919C                 push    ebx             ; hModule
CODE:0040919D                 call    GetProcAddress_0
CODE:004091A2                 mov     [ebp+var_14], eax
CODE:004091A5                 push    offset aVaultgetitem ; "VaultGetItem"
CODE:004091AA                 push    ebx             ; hModule
CODE:004091AB                 call    GetProcAddress_0
CODE:004091B0                 mov     [ebp+var_18], eax
CODE:004091B3                 push    offset aVaultgetitem ; "VaultGetItem"
CODE:004091B8                 push    ebx             ; hModule
CODE:004091B9                 call    GetProcAddress_0
CODE:004091BE                 mov     [ebp+var_1C], eax
CODE:004091C1                 push    offset aVaultfree ; "VaultFree"
```

[21] ShawnFa. "The Differences Between Rijndael and AES". October 2006.

[22] Wikipedia. "PBKDF2".

[23] Microsoft. "Rfc2898DeriveBytes Class".

Sample Timeline

We identified 17 droppers during this research, including:

- 9 samples compiled on 2013-02-17 07:00
- 4 samples compiled on 2009-07-13 23:42
- 3 sample compiled on 2013-10-14 06:48
- 1 sample compiled on 2013-10-13 09:56

The 2009 compile time appears to have been forged, while the 2013 compile times may be legitimate.

In some cases, we found an implant but not the parent dropper. In total, 22 of the 23 implants that we identified during our research had unique compile times ranging from 2013-10-29 until 2014-03-15. We identified two implants that were both compiled on 2014-3-15 at 23:16. These compile times appear to be legitimate and coincide with attempted intrusion activity attributed to these attackers.

Spoofed Installers

Many of the malicious executables (droppers) that we collected were bundled with legitimate installers for VPN or proxy software. Examples include:

- 6dc7cc33a3cdcfee6c4edb6c085b869d was bundled with an installer for Ultrasurf Proxy software.
- 3d26442f06b34df3d5921f89bf680ee9 was bundled with an installer for Gerdoovpn virtual private network software.
- 3efd971db6fbae08e96535478888cff9 was bundled with an installer for the Psiphon proxy.
- 288c91d6c0197e99b92c06496921bf2f was bundled with an installer for Proxifier software.

These droppers were also designed to visually spoof the appearance of the above applications. These droppers contained icons used in the legitimate installers for these programs.

Figure 7: Icon for the Psiphon Anti-censorship Tool

Process Debug (PDB) Strings

Analysis of the PDB strings seen in the implants indicates that there may be more than one developer working on the source code for the Stealer builder. The following two PDB paths were seen in the collection of implants that we collected:

- d:\svn\Stealer\source\Stealer\Stealer\obj\x86\Release\Stealer.pdb
- f:\Projects\C#\Stealer\source\Stealer\Stealer\obj\x86\Release\Stealer.pdb

These strings indicate that the Stealer source code was stored in two different paths but not necessarily on two different computers. The f:\Projects\ path may be from an external storage device such as a thumb drive. It is therefore possible that only one person has access to the source code, but keeps a separate repository on an external storage device. Alternatively, the different file paths could be the result of two different actors storing their source code in two different locations.

Builder Artifacts

In nine of the implants that we collected, we found a consistent portable executable (PE) resource with a SHA256 of 5156aca994ecfcb40458ead8c830cd66469d5f5a031392898d323a8d7a7f23d3. This PE resource contains the VS_VERSION_INFO. In layman's terms, this can best be described as the metadata describing the executable file. This specific PE resource contained the following information:

Note the InternalName of 'Stealer.exe'. This is the attackers' name for this malware family.

```
VS_VERSION_INFO
VarFileInfo
Translation
StringFileInfo
000004b0
Comments
Process for Windows
CompanyName
Microsoft
FileDescription
Process for Windows
FileVersion
1.0.0.0
InternalName
Stealer.exe
LegalCopyright
Copyright
   2013
OriginalFilename
Stealer.exe
ProductName
Process for Windows
ProductVersion
1.0.0.0
Assembly Version
1.0.0.0
```

The "Stealer" Builder and Tools

During our research, we recovered two different tools used by the members of the Ajax Security Team in conjunction with targeted intrusion activities. The first tool, labeled the 'Stealer Builder' was compiled on 2014-04-08. This compile date may indicate that the group is still active.

Upon executing the 'Stealer Builder' the user is presented with an option to load the 'Builder' or to 'Decrypt' logs generated from a victim and exfiltrated to a command-and-control (CnC) server under the groups' control.

The Builder option enables an attacker to configure a new Stealer backdoor. The user can configure the new backdoor to connect to a specific CnC server with a personalized username and password. The attacker can bind the backdoor to a legitimate application of his or her choosing, or they can cloak it with an icon designed to make the backdoor appear as though it is a legitimate file. We also noted that the Builder did not allow the attacker to select a new passphrase or salt used to encrypt the stolen data. The passphrase 'HavijeBaba' and a salt of 'salam!*%#' are both hardcoded into the builder.

Figure 8: The Stealer Tool

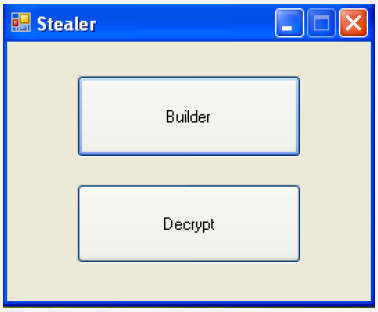
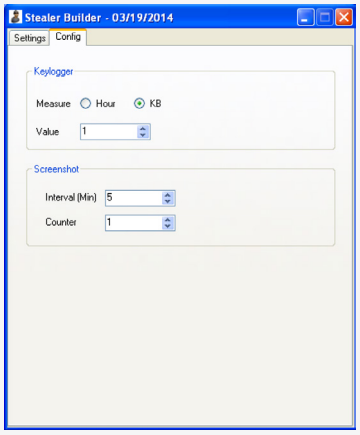
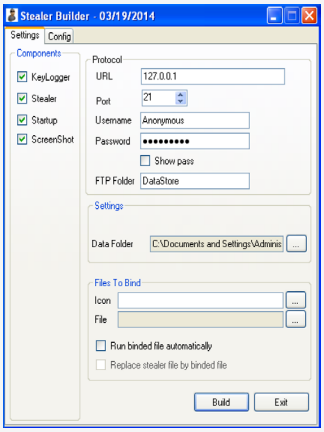


Figure 9: The Stealer Builder





FireEye

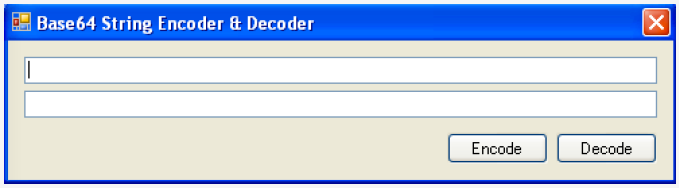
During testing, we observed that backdoors generated by this Stealer Builder had a timestamp of 2013-12-19. We had one backdoor in our repository with this same timestamp. This sample

(MD5 1823b77b9ee6296a8b997ffb64d32d21) was configured to exfiltrate data to ultrasms[.]ir. The VS_VERSION_INFO PE resource mentioned above (SHA256 5156aca994ecfcb40458ead8c830cd66469d5f5a031392898d323a8d7a7f23d3) is an artifact of the Stealer builder that we recovered. The builder generates an executable named IntelRapidStart.exe. This executable contains the aforementioned VS_VERSION_INFO PE resource.

We also recovered a tool designed to encode plaintext into Base64 encoded text or decode Base64 encoded text into plaintext. Members of the Ajax Security Team likely this use tool to encode the configuration data seen in RapidStartTech.stl files. As noted above, the RapidStartTech.stl contains the backdoor's FTP credentials, implant name, decoy name, and screenshot interval, along with boolean settings for startup, keylogger, and screenshot plugins.

Encoding and decoding Base64 data is a straightforward task and the standard Linux operating system offers a number of command line tools to achieve this task. The presence of a Windows-based GUI tool that simplifies encoding and decoding Base64 data indicates that these tools may have been developed for less adept users.

Figure 10: Base64 Encoder



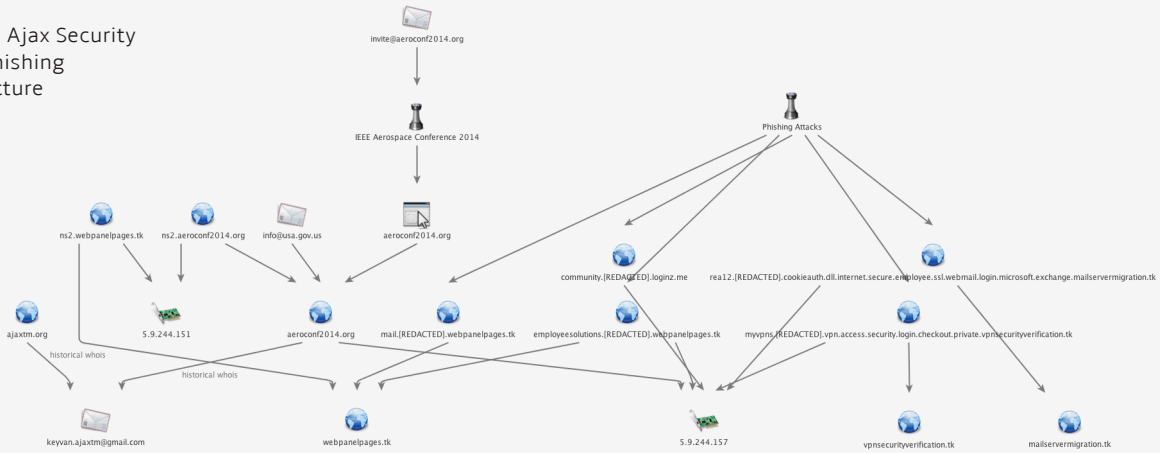
Command-and-Control Infrastructure

The CnC infrastructure consists of distinct, but linked, clusters that have targeted both the users of anti-censorship tools in Iran as well as defense contractor companies in the U.S.

The first cluster contains the domain used in the Aerospace Conference attack as well as the domains used in phishing attacks designed to capture user credentials:

The website used in the Aerospace Conference attack was aeroconf2014[.]org, which is registered to info@usa.gov[.]us. However, historical WHOIS information shows that the domain was registered by keyvan.ajaxtm@gmail[.]com—the same domain used to register ajaxtm[.]org, the website of the Ajax Security Team. The same email addresses were used to register variations of domain names associated with popular services provided by companies such as Google, Facebook, Yahoo and LinkedIn.

Figure 11: Ajax Security Team's Phishing Infrastructure



The second cluster comprises the CnC infrastructure used in the anti-censorship attacks. The majority of the samples we analyzed connect to intel-update[.]com and update-mirror[.]com, which were registered by james.mateo@aim[.]com. The domain intel-update[.]com resolved to the IP address 88.150.227.197, which also hosted domains registered by osshom@yahoo[.]com, many of which are consistent with the pattern of registering domains with associations to Google and Yahoo services. We also observed crossover with a sample that connected to both intel-update[.]com and ultrasms[.]ir, which was registered by lvlr98@gmail[.]com.

Figure 12: Ajax Security Team's Stealer CnC Infrastructure

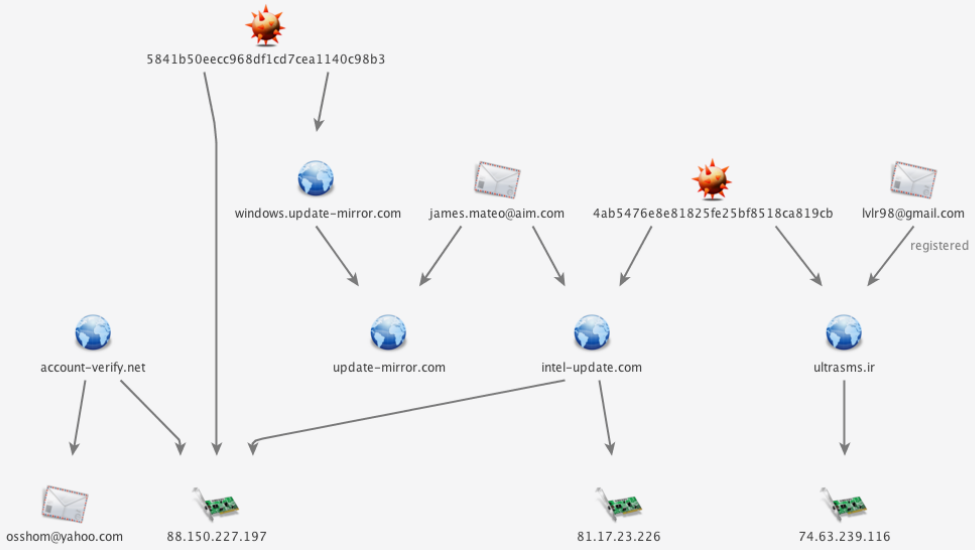
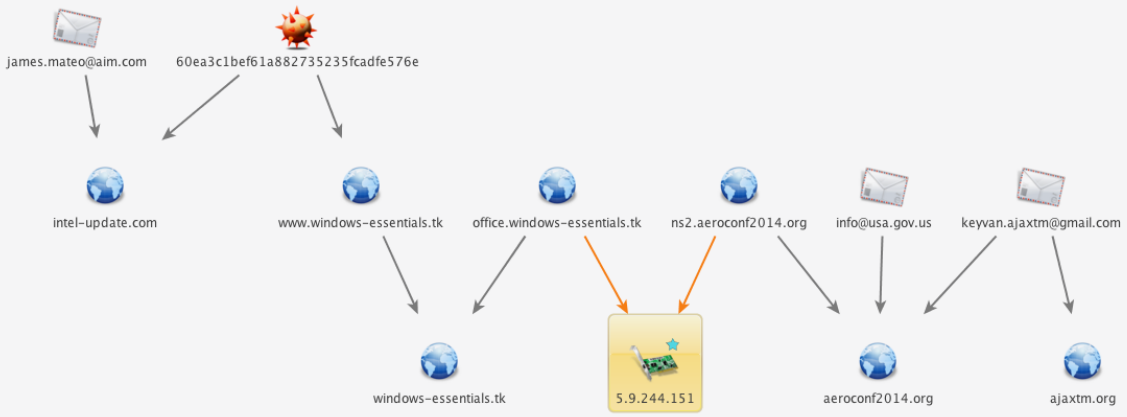


Figure 13: Overlap between the phishing and stealer clusters



These two clusters are linked by a common IP address (5.9.244.151), which is used by both ns2.aeroconf2014[.]org and office.windows-essentials[.]tk.

A third cluster of activity was found via analysis of 1d4d9f6e6fa1a07cb0a66a9ee06d624a. This sample is a Stealer variant that connects to the aforementioned intel-update[.]com as well as plugin-adobe[.]com. The domain plugin-adobe[.]com resolved to 81.17.28.235. Other domains seen resolving to IP address nearby include the following:

Aside from the sample connecting to plugin-adobe[.]com, we have not discovered any malware connecting to these domains.

Victimology

During our investigation, we were able to recover information on 77 victims from one CnC server that we discovered while analyzing malware samples that were disguised as anti-censorship tools. While analyzing the data from the victims, we found that the majority had either their timezone set to "Iran Standard Time" or had their language setting set to Persian:

- 44 had their timezone set to "Iran Standard Time" (37 of those also have their language set to Persian)
- Of the remaining 33, 10 have Persian language settings
- 12 have either Proxifier or Psiphon installed or running (all 12 had a Persian language setting and all but one had their timezone set to "Iran Standard Time")

The largest concentration of victims is in Iran, based on the premise that Persian language settings and "Iran Standard Time" correlate the victim to be geographically located in Iran. As such, we believe that attackers disguised malware as anti-censorship tools in order to target the users of such tools inside Iran as well as Iranian dissidents outside the country.

Domain	IP	First Seen	Last Seen
yahoomail.com.co	81.17.28.227	2013-11-28	2014-4-10
privacy-google.com	81.17.28.229	2014-02-14	2014-02-23
xn--google-yri.com	81.17.28.229	2013-12-08	2014-01-15
appleid.com.co	81.17.28.231	2014-02-20	2014-02-20
accounts-apple.com	81.17.28.231	2013-12-31	2014-02-20
users-facebook.com	81.17.28.231	2014-01-15	2014-01-15
xn--facebook-06k.com	81.17.28.231	2013-11-27	2014-03-07

Attribution

The Ajax Security Team appears to have been formed by personas named "HUrr!c4nE!" and "Cair3x" in 2010.[24] Both members were engaged in website defacements prior to the forming of the Ajax Security Team, and both were members of Iranian hacker forums such as ashiyane[.]org and shabgard[.]org. Other members include "0day", "Mohammad PK" and "Crim3r". The Ajax Security Team website at ajaxtm[.]org had a Web forum with at least 236 members. The group published several exploits for content management systems and engaged in defacements.[25] Initially, the defacements seemed to be motivated by a desire to demonstrate the group's prowess—they even defaced an Iranian government website.[26]

However, the group appears to have become increasingly political. For example, in a blog post in 2012, "Cair3x" announced the targeting of Iran's political opponents.

Figure 14: Cair3x's original blog post and translation

Hacking anti-revolution political and opposition websites

Hello to everyone, After a while of operating underground and enhancing our company's projects and getting close to 24 June 2012, and the martyrdom of Ayatollah Dr. Beheshti and 72 of Imam Khomeini's (First and Former supreme leader of Iran) followers, we have planned a project/ initiative to attack anti-revolution and political websites against the Islamic Republic. And in late hours of Wednesday, June 24, 2012, we attacked these websites and defaced them by writing the words "We are young but we can" on their websites. This is so the enemies of this country know that the blood of our martyr will never be in vain and they will always be remembered in the heart of gallant Iranians.

[24] By March 2010 HUrr!c4nE! was identifying as a member of Ajax Security Team in exploit releases http://www.exploit-db.com/exploits/17011/ and the first defacement archived by Zone-H, which lists both HUrr!c4nE! and Cair3x as members was December 2010 http://www.zone-h.org/mirror/id/12730879

[25] http://osvdb.org/affiliations/1768-ajax-security-team http://www.exploit-db.com/author/?a=3223 http://packetstormsecurity.com/files/author/9928/

[26] http://www.zone-h.org/mirror/id/13225183

In 2013, the Ajax Security Team, and "HUrr!c4nE!" in particular, took part in "#OpIsrael" and "#OpUSA".27

By early 2014, the Ajax Security Team appears to have dwindled. There have been no defacements since December 2013. The website and forum at ajaxtm[.]org operated by "HUrr!c4nE!", aka "k3yv4n", is no longer active.

"HUrr!c4nE!" has the most open/documented Internet persona of the Ajax Security Team. He registered the ajaxtm[.]org domain name using the email address keyvan.ajaxtm@gmail[.]com. This was also the email address used to register the domain aerospace2014[.]org, which was used in spear phishing attacks against companies in the U.S. and is linked with malware activity directed at users of anti-censorship tools in Iran.

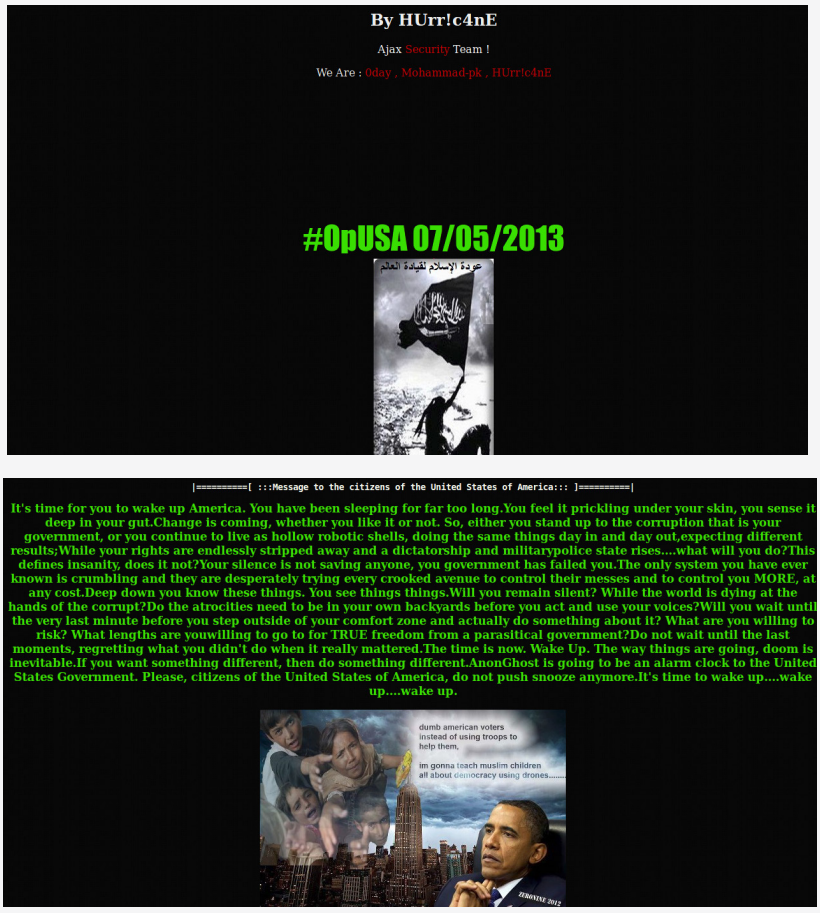


Figure 15: Screenshot of the defacement content used in #OpUSA

[27] Ashraf, N. "#OpIsrael: Hacktivists Starting Cyber Attack against Israel on 7th of April". March 2013. "OpUSA Targeting Government & Financial Sectors on 07 May 2013: Likely Tools, Targets and Mitigating Measures". May 2013.

"HUrr!c4nE!" features prominently in all the group's activities and defacements. Although there has been a decline in public-facing Ajax Security Team activity, this coincides with an increase in malware activity linked to the group's infrastructure.

• ~2009—Membership in ashiyane.org and shabgard.org forums

• 2010 – 2012—Defacements, Release of exploits for CMS

• 2012 – 2013—Increasing politicization, participation on #OpIsrael, #OpUSA

• 2013 – 2014—Transition to cyber-espionage

The increasing politicization of the Ajax Security Team aligns with the timing of their activities against the perceived enemies of Iran. In addition to attacking companies in the U.S., they have targeted domestic users of anti-censorship technology.

While the objectives of this group are consistent with Iran's efforts at controlling political dissent and expanding offensive cyber capabilities, the relationship between this group and the Iranian government remains inconclusive.

For example, the Ajax Security Team could just be using anti-censorship tools as a lure because they are popular in Iran, in order to engage in activities that would be considered traditional cybercrime. In one case, "HUrr!c4nE!", using the email address keyvan.ajaxtm@gmail[.]com, has been flagged for possible fraud by an online retailer. While "HUrr!c4nE!" is engaged in operations that align with Iran's political objectives, he may also be dabbling in traditional cybercrime.

This indicates that there is a considerable grey area between the cyber espionage capabilities of Iran's hacker groups and any direct Iranian government or military involvement.

On the spectrum of state responsibility, these attacks align with state-encouraged attacks, which are defined as attacks in which:

Third parties control and conduct the attack, but the national government encourages them as a matter of policy.[28]

Recruiting hackers through this model allows Iran to influence their activities, and provides the Iranian government plausible deniability, but a lack of direct control also means that the groups may be unpredictable and engage in unsanctioned attacks.

Figure 16: Screenshot of an online retailer's fraud alert

OrderID	Customers ID	IP	E-mail	Shipping Address & ZIP	Customer Name	CC number	Payment Method	Expiration Date	Source	Created
140217MYSJ93	1957612	176.67.169.223 - 82.220.3.101	keyvan.ajaxtm@gmail.com	jliuktuik-Denver-CO-United States		401795XXXXXX0711	paypalwpp_cc	0514	Fraud order database	2014-02-17 23:30:18

[28] Healey, J. "Beyond Attribution: Seeking National Responsibility for Cyber Attacks". January 2012.

Conclusion

The increased politicization of the Ajax Security Team, and the transition from nuisance defacements to operations against internal dissidents and foreign targets, coincides with moves by Iran aimed at increasing offensive cyber capabilities. While the relationship between actors such as the Ajax Security Team and the Iranian government is unknown, their activities appear to align with Iranian government political objectives.

The capabilities of the Ajax Security Team remain unclear. This group uses at least one malware family that is not publicly available. We have not directly observed the Ajax Security Team use exploits to deliver malware, but it is unclear if they or other Iranian actors are capable of producing or acquiring exploit code.

While the Ajax Security Team's capabilities remain unclear, we know that their current operations have been somewhat successful as measured by the number of victims seen checking into to an Ajax Security Team controlled CnC server. We believe that if these actors continue the current pace of their operations they will improve their capabilities in the mid-term.

About FireEye

FireEye has invented a purpose-built, virtual machine-based security platform that provides real-time threat protection to enterprises and governments worldwide against the next generation of cyber attacks. These highly sophisticated cyber attacks easily circumvent traditional signature-based defenses, such as next-generation firewalls, IPS, anti-virus, and gateways. The FireEye Threat Prevention Platform provides real-time, dynamic threat protection without the use of signatures to protect an organization across the primary threat vectors and across the different stages of an attack life cycle.

The core of the FireEye platform is a virtual execution engine, complemented by dynamic threat intelligence, to identify and block cyber attacks in real time. FireEye has over 1,500 customers across more than 40 countries, including over 100 of the Fortune 500.

We thank Kenneth Geers and Jen Weedon for their support and analysis on these findings.

FireEye, Inc. | 1440 McCarthy Blvd. Milpitas, CA 95035 | 408.321.6300 | 877.FIREEYE (347.3393) | info@fireeye.com | **www.fireeye.com**