# Connecting the Dots: Syrian Malware Team Uses BlackWorm for Attacks

fireeye.com/blog/threat-research/2014/08/connecting-the-dots-syrian-malware-team-uses-blackworm-for-attacks.html

## Threat Research Blog

August 29, 2014 | by Kyle Wilhoit, Thoufique Haq

The Syrian Electronic Army has made news for its recent attacks on major communications websites, Forbes, and an alleged attack on CENTCOM. While these attacks garnered public attention, the activities of another group - The Syrian Malware Team - have gone largely unnoticed. The group's activities prompted us to take a closer look. We discovered this group using a .NET based RAT called BlackWorm to infiltrate their targets.

The Syrian Malware Team is largely pro-Syrian government, as seen in one of their banners featuring Syrian President Bashar al-Assad. Based on the sentiments publicly expressed by this group it is likely that they are either directly or indirectly involved with the Syrian government. Further certain members of the Syrian Malware Team have ties to the Syrian Electronic army (SEA) known to be linked to the Syrian government. This indicates that the Syrian Malware Team may also be possibly an offshoot or part of the SEA.


Banner used by the Syrian Malware Team

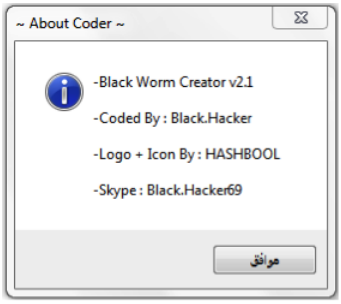
## BlackWorm Authorship

We found at least two distinct versions of the BlackWorm tool, including an original/private version (v0.3.0) and the Dark Edition (v2.1). The original BlackWorm builder was co-authored by Naser Al Mutairi from Kuwait, better known by his online moniker 'njq8'. He is also known to have coded njw0rm, njRAT/LV, and earlier versions of H-worm/Houdini. We found his code being used in a slew of other RATs such as Fallaga and Spygate. BlackWorm v0.3.0 was also co-authored by another actor, Black Mafia.


About section within the original version of BlackWorm builder

Within the underground development forums, it's common for threat actors to collaborate on toolsets. Some write the base tools that other attackers can use; others modify and enhance existing tools.

The BlackWorm builder v2.1 is a prime example of actors modifying and enhancing current RATs. After njq8 and Black Mafia created the original builder, another author, Black.Hacker, enhanced its feature set.



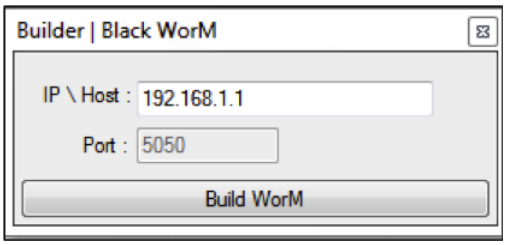

About section within BlackWorm Dark Edition builder


Black.Hacker's banner on social media

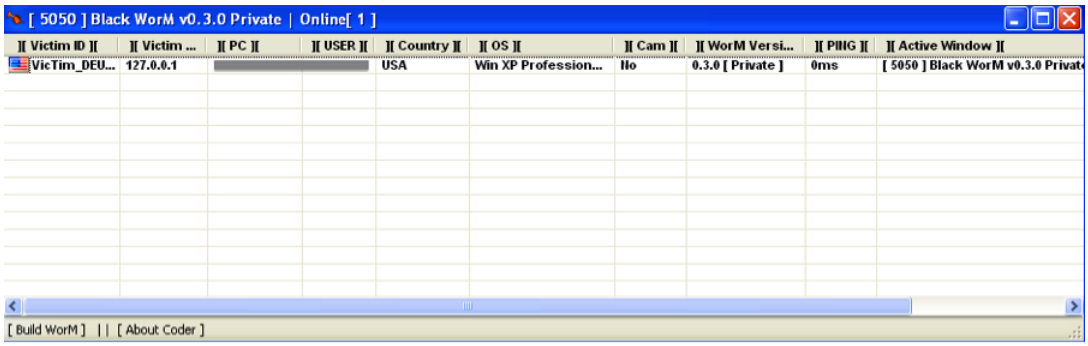

As an interesting side note, 'njq8' took down his blog in recent months and announced a cease in all malware development activity on his Twitter and Facebook account, urging others to stop as well. This is likely a direct result of the lawsuit filed against him by Microsoft.

## BlackWorm RAT Features

The builder for BlackWorm v0.3.0 is fairly simple and allows for very quick payload, but doesn't allow any configuration other than the IP address for command and control (C2).




Building binary through BlackWorm v0.3.0




BlackWorm v0.3.0 controller

BlackWorm v0.3.0 supports the following commands between the controller and the implant:

| | |
|---|---|
| ping | Checks if victim is online |
| closeserver | Exits the implant |
| restartserver | Restarts the implant |
| sendfile | Transfer and run file from server |
| download | Download and run file from URL |
| ddos | Ping flood target |
| msgbox | Message interaction with victim |
| down | Kill critical windows processes |
| blocker | Block specified website by pointing resolution to 127.0.0.1 |
| logoff | Logout out of windows |
| restart | Restart system |
| shutdown | Shutdown system |
| more | Disable task manager, registry tools, system restore. Also blocks keyboard and mouse input |
| hror | Displays a startling flash video |

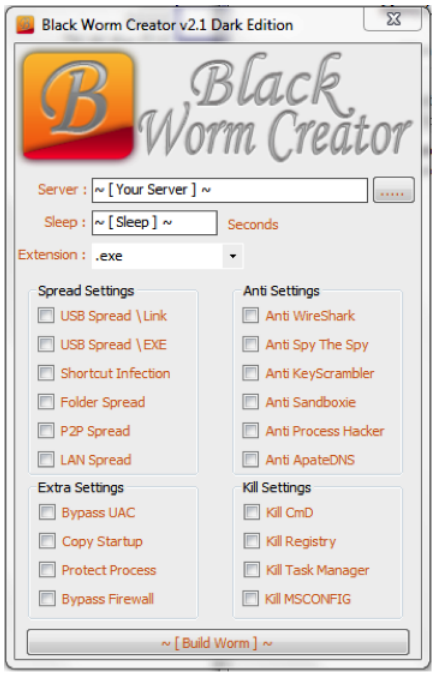
In addition to the features supported by the command structure, the payload can:

- Seek and kill no-ip processes DUC30 and DUC20
- Disable Task Manager to kill process dialog
- Copy itself to USB drives and create autorun entries
- Copy itself to common peer-to-peer (P2P) share locations

- Collect system information such as OS, username, hostname, presence of camera, active window name, etc., to display in the controller
- Kill the following analysis processes (if found):
    - procexp
    - SbieCtrl
    - SpyTheSpy
    - SpeedGear
    - Wireshark
    - MBAM
    - ApateDNS
    - IPBlocker
    - cPorts
    - ProcessHacker
    - AntiLogger

The Syrian Malware Team primarily uses another version of BlackWorm called the Dark Edition (v2.1). BlackWorm v2.1 was released on a prolific underground forum where information and code is often shared, traded and sold.


BlackWorm v2.1 has the same abilities as the original version and additional functionality, including bypassing UAC, disabling host firewalls and spreading over network shares. Unlike its predecessor, it also allows for granular control of the features available within the RAT. These additional controls allow the RAT user to enable and disable features as needed. Binary output can be also be generated in multiple formats, such as .exe, .src and .dll.




BlackWorm Dark Edition builder

## Syrian Malware Team

We observed activity from the Syrian Malware Team going as far back as Jan. 1, 2011. Based on Facebook posts, they are allegedly directly or indirectly involved with the Syrian government. Their Facebook page shows they are still very active, with a post as recent as July 16th, 2014.

Syrian Malware Team's Facebook page

The Syrian Malware Team has been involved in everything from profiling targets to orchestrating attacks themselves. There are seemingly multiple members, including:

https://www.facebook.com/hawk.syrian.9

https://www.facebook.com/kays.syr

Partial list of self-proclaimed Syrian Malware Team members

Some of these people have posted malware-related items on Facebook.


Facebook posting of virus scanning of files

While looking for Dark Edition samples, we discovered a binary named *svchost.exe* (MD5: 015c51e11e314ff99b1487d92a1ba09b). We quickly saw indicators that it was created by BlackWorm Dark Edition.

```
public static string h = "[redacted]";
public static int port = Conversions.ToInteger("5050");
public static string meltf = "False";
public static string Name = "12121212";
public static string Y = "/j|n\\";
public static string Ver = "2.4.0 [ Dark Edition ]";
public static string uexe = "False";
public static string ulink = "False";
public static string up2p = "False";
public static string startUP = "5bd2b23cd2234db4ebd4";
public static string BD = "False";
public static string exen = "Server.exe";
public static string firewall = "False";
public static string ByUAC = "False";
public static string SkypeSpread = "False";
public static string Mag = "";
public static string FolderSpread = "False";
```

Configuration options within code

The malware communicated out to 178.44.115.196, over port 5050, with a command structure of:

!0/j|n\12121212_64F3BF1F/j|n\{Hostname}/j|n\{Username}/j|n\USA/j|n\Win 7 Professional SP1 x86/j|n\No/j|n\2.4.0 [ Dark Edition]/j|n\/j|n\{ActiveWindowName}/j|n\[endof]

When looking at samples of Dark Edition BlackWorm being used by the Syrian Malware Team, the strings “Syrian Malware,” or “Syrian Malware Team” are often used in the C2 communications or within the binary strings.

Additional pivoting off of *svchost.exe* brought us to three additional samples apparently built with BlackWorm Dark Edition. E.exe, (MD5: a8cf815c3800202d448d035300985dc7) a binary that drew our attention, looked to be a backdoor with the Syrian Malware strings within it.

```
aliallosh.sytes.net
Syrian Malware
Restart
Microsoft
Windows
[endof]
ToArray
Length
```

When executed, the binary beacons to aliallosh.sytes.net on port 1177. This C2 has been seen in multiple malware runs often associated with Syria.  The command structure of the binary is:

!0/j|n\Syrian Malware/j|n\{Hostname}/j|n\{Username}/j|n\USA/j|n\Win 7 Professional SP1 x86/j|n\No/j|n

\0.1/j|n\/j|n\{ActiveWindowName}/j|n\[endof]

Finally, pivoting to another sample, 1gpj.srcRania (MD5:f99c15c62a5d981ffac5fdb611e13095), the same strings were present. The string "Rania" used as a lure was in Arabic and likely refers to the prolific Queen Rania of Jordan.

```
_CorExeMain
mscoree.dll
syrian Malware
AppData
Temporary Projects
ali2.pdb
```

The traffic is nearly identical to the other samples we identified and tied to the Syrian Malware Team.

!1/j|n\C:\Documents and Settings\{Username}\Local Settings\Application DataldoDrZdpkK.jpg - Windows Internet Explorer[endof]!0/j|n\Syrian Malware/j|n\{Hostname}/j|n\{Username}/j|n\USA/j|n\Win XP ProfessionalSP2 x86/j|n\No/j|n\0.1/j|n\/j|n\C:\Documents and Settings\{Username}\Local Settings\Application DataldoDrZdpkK.jpg - {ActiveWindowName}/j|n\[endof]

## Conclusion

Determining which groups use which malware is often very difficult. Connecting the dots between actors and malware typically involves looking at binary code, identifying related malware examples associated with those binaries, and reviewing infection vectors, among other things.

This blog presents a prime example of the process of attribution. We connected a builder with malware samples and the actors/developers behind these attacks. This type of attribution is key to creating actionable threat intelligence to help proactively protect organizations.

Previous Post
Next Post