Crisis procedure

- Keep VirusTotal monitored (**Antonio, Daniele, Guido**)

To be done always (also for exploits leaks)

When a leak is detected:

Criteria:

- sample from VT automatically received on @vt
- sample out-of-band forward to @vt
- article sent to @media
- 0day advisory forward to @vt
- ticket automatically received on @rcs-support

Actions:

- verify the threat level of the leak
  - sample: unpack, find version, client, synch address, factory ID (**Guido, Antonio, Zeno, Que**) [make all the cases]
  - address: find watermark, forward to @vt
  - exploit: check if signature is published, forward to @vt
- associate watermark to client
- kill anonymizer from the killer VPS
- send the "Leaked address" (macro) ticket for confirmation
- once confirmed, send "Leaked address/exploit confirmed" or "Leaked agent confirmed" according to the specific case
  - 1. open the Console, go to the System -> Frontend panel, open the File Manager, Select all the entries and Delete
  - 2. on the Collector, browse to c:\RCS\Collector\public and delete any residual file
  - 3. Shutdown the Collector
  - 4. Close (DO NOT DELETE) the <factory ID> factory and any suspect agent.
  - 5. Save the latest DEVICE evidence for each of the agents of factory <factory ID>, make a zip and send it to us. Inform us on the infection vector (e.g. exploits, java applet) used with the factory.

MACRO (news): exploit is patched, unsafe to use

Procedure to recover agents (TBD after few days after collector shutdown)

- entitle a FAE to follow the customer (**Daniele**)

MACRO: add a new anonymizer

- reconfigure existing agents to sync against the new anon
- exception: if agents cannot be moved (marked as bad), move them back on the chain, avoiding the collector
- check the new config
- burn the VPS
- if a new version is released to fix the leak, delay the release to the bastard until the above procedure is completed

R&D

When testing detects a new signature or latest version leaks or exploit leaks:

- If exploit is leaked (**ALoR**)
    - Remove any compromised exploit from the package(**ALoR**)
    - Release a new exploit package that requires a new RCS version (**ALoR**)
        - The new RCS version package must erase all the files in \public folder
    - Modify shellcode and first-stages to evade AVs (**Guido**)
- If exploit and/or latest version is leaked
    - Format the VPS used for delivering the exploits. After few days, if the exploit has not been patched, restart the exploit service using different VPSs (**Guido**)
        - Give to all the customers that have the full exploit pack a new VPS-relay for exploit delivery
- If the exploit VPS is not formatted, and we release a patch for the scout: stop the VPS for a few days, delete all the pending exploits and ask the customers to create the silent installers again (after installing the patch) (**Guido**)
- Follow the specific case for the type of leak
- If two simple leaks (scout or elite) verifies close together, refer to "Total crisis"

CONSOLE LEAK

PROCEDURE

Support

- Get in touch with the customer affected. Check on the Audit system if the attacker accessed any data. (**Daniele**)

R&D

- Release a new backend version (**Alor**)
- Old consoles cannot be upgraded
- New server only accepts connections from new consoles (by changing a bit the authentication scheme)

SCOUT LEAK

CRITERIA

- If a sample is public and the version is new (not leaked)
- If the automated testing system verifies an AV signature

PROCEDURE

Support

- MACRO: issue a News to avoid upgrading scout to elite and avoid new infections. Already installed agents should be OK. Wait for the new release. (**Daniele**)

R&D

- Update the scout (**Guido**)
    - Update the packer to bypass signatures
    - Update version ID **(Soldier too)**
    - Update executable names, icons and product descriptions in the resources.
    - Update the User Agent for URL requests (sync and social) **(Soldier too x 2)**
    - Update exported symbol name (e.g. MyConf)
    - Update URL for sync request (e.g. index.php) **(Soldier too)**
    - Change shared memory's name length **(Soldier according to Scout)**
    - Change .bat's name length
    - Change fake message strings
    - Verify if the melted structure is compromised, eventually update (Scout and Network Injector)
    - Adopt a new certificate
    - Change packer executables in the zipped core file (es: VMProtect_Con.exe in windows.zip)
- Update the server (**ALoR**)
    - Block upgrades of existing scouts to elite
        - by verifying scout version ID
        - this step must be done in all cases (Scout and Total Crisis)
    - The new RCS installer must erase all the files in the \public folder
- Update the scripts for exploit creation on the VPS to check for the latest scout (**Guido**)
    - All the new exploits must be created using the latest scout
    - Remove all the pending exploits from the EDN
- Have a new certificate issued (**Vale**)
- Check if there is any signature available for the protocol (eg: snort) (**Guido**)

Note: **If it's a minor leak, only underlined countermeasures; then perform a "Scout Manual Testing"**

ELITE LEAK

CRITERIA

- If a sample is public and the version is new (not leaked)
- If the automated testing system verifies a AV signature

PROCEDURE

Support

- MACRO (news): avoid any upgrade

R&D

- Update strings in .rsrc of core32 (version, program name, etc.) (**Marco**)
- Change the version ID (YYYYMMDDPP) (**Marco**)
- Update URL for sync request with different name than Scout (e.g. index.php) (**Marco**)
- Update registry key names (**ALoR, Marco**)
    - Keep tracks of old and new keys in the backend for offline installation
    - Keep uniqueness with random numbers (max 23 chars, including numbers)
    - Make the registry startup command line non fingerprintable changing the environment variables (in HM_InsertRegistryKey)
    - Check new names with Kaspersky to avoid alarms
- Update the User Agent for URL requests (both in Sync and Social Module) (**Marco**)
- Update 32/64 bit DLL packer (**Antonio**)
    - If some AV cannot be bypassed, add it to the blacklist (**ALoR**)
- Remove compromised files from windows.zip: check also AV tests for codec & SQLite (**Marco**)
- Update the offline cd if the drop directory is changed (**Marco**) - only in extreme cases
- Check if there is any signature available for the protocol (eg: snort) (**Guido**)
- Update the server to prohibit upgrade from old to new elites (**ALoR**)
    - In case of Total Crisis, this check could be redundant since:
        - "BAD" marker inhibits upgrades of instances marked as bad, and inhibits creation of new instances from factories marked as bad.
- If the scout didn't leak (but is in danger) :
    - Change the scout certificate (**ALoR**)
    - Request a new certificate (**Vale**)
    - Change scout version ID (**Guido**)
    - Inhibits upgrade to elite for older scouts (**ALoR**)
        - Less risk of elite leaks!
        - For special situations on some target check case by case

LEAK of OSX, Linux or Mobile

- Update the code to bypass the signatures [compromised platform developer]
- Update the server to avoid upgrade of compromised agents (**ALoR**)
- Evade any existing yara on OSX:
    - Insert VMProtect
    - Or simply change few bytes at the beginning of the dropper (the whole code should change due to native encryption)

SCOUT and ELITE ("Total crisis") or close leaks

Support

MACRO: update communication according to current crisis (**Daniele**)

- evaluate if global collector shutdown is needed
    - clean public folder
- MARK_BAD and new anonymizers could be not necessary. Evaluate the situation.
    - If new anonymizers are provided to the customers: Customers that are using DNS names must change these names with something new and not related to their country.
    - Set a firewall rule on the new anonymizers' 443 port: accept connections from Collectors ONLY.
    - Check if firewall rules allow connections to Collectors from Anonymizers ONLY.
    - Check if the Backend server is not publicly reachable from the internet.
    - Check if the installation is all-in-one. Deny IT!
    - Dismiss any old (bad) anonymizer that is no longer in use.

- wait for R&D to issue new release
- change all the customers' watermarks (**Daniele**)
- issue new licenses (give to customers during "Procedure to add new anonymizers")
- tell Customer to upgrade to elite any existing relevant scout (they cannot be upgraded after)
- add safe anonymizers on top of compromised chains (**Daniele, FAE, ALoR**)
  MACRO: ask for new anonymizers (never been used)
    - new anonymizers must be created AFTER the upgrade

ref to "Procedure to add new anonymizers"

R&D

- follow both scout and elite leak R&D procedures
- force update of exploit pack (new version) (**ALoR**)
- if MARK_BAD is inserted, change the decoy page to make fingerprinting more difficult(**ALoR**)
    - Change the decoy pages' array (eg: Server Error)
    - Change default replies made by Ruby class (put NGINX in front)
    - The new decoy pages must be served ONLY to requests made through "good" anonymizers (old anonymizers cannot be used to find the new decoy!)
- insert MARK_BAD in the installer if needed (**ALoR**)
    - all existing factories, instances and anonymizers are marked bad:
        - FACTORY:
            - Cannot be built
            - New instances cannot be created (decoy page returned)
        - INSTANCE:
            - Reconfiguration is allowed excluding GOOD anonymizers
            - Synchronizations through a good anonymizer is prohibited (version field in anonymizer s headers marks it as good to prevent anon fingerprinting; decoy page returned)
            - Cannot be upgraded
        - ANONYMIZER:
            - Cannot be placed in front of a good (blocked from Console)
            - Cannot be upgraded
            - GOOD factory cannot be built synchronizing to a BAD anon
            - GOOD instance cannot be reconfigured synchronizing to a BAD anon

  Update the anonymizer's version considered good with the release Version ID (**ALoR**, **Fabio**)

Potential one shot improvements (to be evaluated)

- Dedicate, of the two additional anonymizers, one exclusively for synchronizing elite agents, and the other for all the remaining platforms.

TEST

TODO: ask Zeno, Guido to define a set of functional tests to include

Continuous testing

Update the build server (Minotauro) with the latest release candidate

- Use a template with all the modules activated from the Basic configuration
- Run the scout executable
- Wait first sync (5 min.)
- Upgrade the agent to elite
- Verify agent is upgraded
- Verify behavior according to blacklist
- Revert the snapshot
- Run the melted executable
- Run the scout just created
- Wait first sync (5 min.)
- Upgrade the agent to elite
- Verify agent is upgraded
- Verify behavior according to blacklist
- Run the executable document (.txt)
- Wait first sync (5 min.)
- Upgrade the agent to elite
- Verify agent is upgraded
- Verify behaviour according to blacklist

Distribute agents in all VM to Desktop\avtest

- scan della cartella Desktop\avtest (agenti Linux, OSX, Mobile, exploit)

Inform the Clients if something changes.

Normal release testing

- Perform a Continuous Testing (extended version); check that every AV is ok
- Generic functional tests (Skype, Social, Hiding, Persistence)
    - dedicated system with no AV, application software installed
- Test all Network Injector infection methods
- Test Offline-cd installation on a blacklisted AV (installation should be inhibited)
- Test the upgrade of the agent from the old version if possible (not in crisis)

Mac

- disconnect network
- install agent with Intego real scan active
- uninstall the agent

Android

- TODO

Crisis release testing

Before testing wait for signatures to be propagated to main AVs

- Perform a Normal Release Testing

- Test scout to elite upgrade + persistence on:
  - XP (test hiding too)
  - Windows 8
  - Windows 7 (Arabic user with low privileges)

- Test Offline-CD (Install/Uninstall) on a non-blacklisted AV

- Check if the elite silent installer (used in the demos) is not detected by Kaspersky

- Test scout-to-elite upgrade on a physical machine with Kaspersky

- Test if the scout (and the soldier) evades VM and Cackoo
  - Use latest versions of Cackoo and VMWare Workstation
  - Install an AV of choice
  - The scout must exit without any action
  - Use a stub of code to verify if the checks evade behavioral analysis on VirusTotal

- Using RC packages check if:
  - Scout and elite have the correct version ID in console
  - Scout has new name, icon and description (if modified)
  - Scout has the new certificate (if modified)
  - Elite uses new registry keys (if modified)
  - Scout and elite(32/64) use latest packers (if modified)

- Manual AV testing on automated test VMs (ref to "AV Test Cycle")
  - Use a template with all the modules activated from the Basic configuration. Uninstall on calc.exe

- If a particular Watermark leaked (and we did not modify all the watermarks), insert this watermark on Minotauro for a test run. If it is detected, modify it for that customer (pay attention if it compromises customer's upgrades)

- Perform upgrade tests. Check if the upgrade is possible or if it is correctly inhibited.
  - NOT on Minotauro
  - Check again certificate, names, packers, version IDs...

- Check if some modification in the agent behavior is required (**Marco, ALoR**)
  - Network Protocol, Access to a given resource, etc.

Scout manual testing

- Double-click on agent.exe
- Verify it has been copied into the startup folder after ca. 15 sec
- Logoff / logon
- Double-click on agent.exe(again), verify it exits immediately
- Wait 5 minutes, move mouse/keyboard, check sync and device evidence(screenshot if enabled)
- Upgrade from the console
- Logon/logoff + wait for 5 minutes + move mouse/keyboard

- If elite:
    - Verify sync (elite)
    - Double-click on agent.exe(again), check if it exits immediately
    - Logon/logoff + check the scout has been delete from startup folder

- If soldier:
    - The startup folder must contain only the soldier (>= 1500kb with different icon/name)
    - Double-click on agent.exe(again), check if it exits immediately
    - Verify the soldier's digital signature
    - Logoff/logon, move mouse/keyboard and check sync
    - Uninstall from console
    - Wait for sync and check soldier has been deleted from startup

- Test scout to elite upgrade on:
    - XP
    - Windows 8
    - Windows 7 (Arabic user with low privileges)

- Executable Document:
    - Create a fake doc from console
    - Execute it, wait for 15 sec
    - Check the scout in the startup folder and his digital signature
    - Check the fake document has been replaced by the original document

- Melted:
    - Melt a putty from the console
    - Double click on it
    - Wait for 15 sec, check the startup folder and verify the digital signature
    - And wait for first sync
    - Uninstall from console
    - Logoff/logon, wait 6 minutes, move mouse/keyboard
    - Wait for the sync and then verify scout has been deleted from startup

- TNI/NIA Melted:
    - Ask Andrea for a melted putty (from his TNI)
    - Double click on it
    - Wait for 15 sec, check the startup folder and verify the digital signature

AV Test - cycle

Concurrently:

- Run the script to distribute samples on the VMs

For each VM:

- Verify Antivirus is up to date
- Run Desktop\AVTEST\build_scout_minotauro.bat
- Scan the scout in Desktop\AVTEST\build, then run it
- Perform a quick scan with the AV and/or scan the Startup directory
- Wait first sync (5 min)
- Upgrade the agent to elite
- Logoff/logon
- Wait sync Elite (6 min), moving the cursor
- Write something in notepad
- Logoff/logon
- Wait sync (1 min)
- Verify that Elite logs arrived (keylog in notepad)
- Logoff/Logon
- Perform a quick scan
- Scan user's home directory
- Check it the Elite continues to sync regularly (wait for at least 2 syncs)
- Uninstall (by running calc.exe)

AV Test - Mac

- disconnect network
- install agent with Intego real scan active
- uninstall the agent

AV Test - Android

- TODO

Conclusion

- Create the Invisibility Report