Endpoint Security Policy System

A Network Access Control System with Vulnerability Detection and User Remediation

Evgueni Martynov

UNIX Systems Group

Mike Wiseman

Computer Security Administration

Table of Contents

Acknowledgements

Change History

Doc Version	Date	Description
1.0	October 23, 2007	Original for ESP v-1.0.
1.1	August 31, 2011	Modifications to server install and addendum for Fedora 15.

Summary

The Endpoint Security Policy (ESP) system is an implementation of Network Access Control (NAC) which is functionality used to protect an institution's computer networks from the connection of unmanaged, user-owned desktop or laptop computers. This is a common scenario in university and college environments in which students attach their desktop or laptop computers to a wired or wireless network at a residence or academic facility.

ESP is an in-house system developed using open source packages and utilities from vendors to provide three components: a network isolation system, a run-once utility which users are asked to run on their Microsoft OS computers to detect the state of critical security updates and antivirus, and user self-remediation to encourage users to repair their own equipment without IT staff intervention.

The system has been in use for over two years at the University of Toronto and, during the course of a term, is used on approximately 25,000 computers. It has been very effective in protecting the University's networks from exploit and compromise incidents as well as helping users to manage some basic aspects of their computers.

Overview

Host computers at higher-ed institutions can be classified in two groups: managed and unmanaged. These groups refer to whether the institution owns and manages the equipment. Managed computers are owned by the institution which usually provides resources for their maintenance. Unmanaged computers are those owned by their user – usually students in residences and those connected to wireless networks. The latter group is responsible for the maintenance of their own equipment but often do not meet this responsibility due to inexperience or technical problems. Nevertheless, it is critical from the perspective of the institution that these computers be maintained.

Many institutions experienced major computer network degradations and outages in 2003-2004 when connected Microsoft OS computers became compromised due to the Slammer and Blaster worms. This malicious code exploited vulnerabilities in the operating systems of the computers – vulnerabilities which could have been eliminated if the installation of updates had been done in a timely manner. Since that time, a technology called Network Access Control has evolved which, in part, deals with ensuring software is updated and antivirus is functioning and up-to-date before full network access rights are granted to the connecting user. There are a number of commercial and open source products which implement NAC – ESP is one of them. It was developed locally due to the availability of detection command line tools. Note that there is no current facility to check for UNIX or other OS vulnerabilities.

There are three components to an ESP system: network isolation, vulnerability detection and user self-remediation. Network isolation is implemented using the well known Netreg software which uses two pools of DHCP addresses – the first providing a zone in which a user has limited access to network services, and the second to provide full network access. The second component, vulnerability detection, is used to detect the status of: 1) operating system and supported application updates, implemented using the Microsoft Baseline Security Analyzer detection tool and 2) antivirus operation and definition up-to-date check done by querying the Windows Management Instrumentation database on the test machine. The third component is user self-remediation. The Microsoft products had built-in capabilities to update themselves as well as user functionality in the form of Windows Update. It was, therefore, a requirement to help users use these maintenance tools to avoid involving institutional help desk or IT staff.

Network Isolation

Network isolation or quarantine zone is implemented in the ESP package using the open source package called NetReg (netreg.sourceforge.net) v-1.3. The basic theory behind Netreg is as follows: Netreg operates using two pools of DHCP addresses, one pool is non-routable, the other routable. It runs under UNIX and contains configuration files to set up the DHCP server as well as web documents and scripts which make up a user and an administrator interface.

Originally, NetReg components were: a DHCP server, a DNS server,and an Apache web server. We replaced DNS server with Squid Internet cache server and make use of Linux IPTables firewall to implement network isolation for non-registered users.

When an unregistered computer (client) connects to the network it gets an non-routable IP address. The Squid and the IPTables firewall are configured to allow traffic to a limited number of domains,to local university hosts to download software, to Microsoft to get the latest updates for Windows OS and applications (MS Office), to AntiVirus sites to get AntiVirus software and updates. Squid Internet cache works as a transparent proxy.

The client gets a page with a link to run vulnerability detection agent (ESP agent) in order to be checked for missing Microsoft patches and status of an AntiVirus software (running, up-to-date).

We would like to mention here that computers with non-Windows OS skip the scan and will be redirected to a Registartion Page where users can provide information needed for registration (name, room number, etc). After a user types in that information the computer will receive a new routable IP address within two minutes.

We use HTTP cookies to communicate the status of client computer to the ESP server in order for the server to display the needed page – the Fail Page in the case of a missing patch or the Registration Page if computer has all the patches and AV is up-to-date. Scan status cookie is written by the ESP Agent to the hard drive and will be late sent by the Internet Explorer browser to the ESP server.

Vulnerability Detection

The heart of the ESP system is the agent or utility which makes use of Microsoft's MBSA security analysis tool to check the status of operating system and application security updates. It also detects the status of antivirus protection using two useful checks – writing an EICAR pattern to see if the AV is functional and a Windows Management Interface (WMI) check to obtain Windows Security Center data on the status of antivirus definition updates. The utility is built using the useful open source package Nullsoft Scriptable Installer System (NSIS) which provides a simple API to develop simple Windows applications.

The administration and configuration of the tool is designed to be done either centrally or departmentally to reflect the de-centralized nature of higher educational institutions. The vulnerability detector can be configured to be required to be run by users at arbitrary times. One way in which detector tests are scheduled is shortly after the Microsoft "patch Tuesday" in which updates are released and occurs once a month. So users are required to run the check once a month to ensure that all appropriate updates are installed in a timely manner.

The following flowchart provides a basic overview:

```
┌─────────────────────────────┐
│           START             │
│                             │
│ -check user for admin rights│
│ -unpack files in temp folder│
│ -read configuration file from│
│ central network location    │
└─────────────────────────────┘
              │
┌─────────────────────────────┐
│      CHECK MICROSOFT         │
│    CRITICAL UPDATE STATUS    │
│                             │
│ -run mbsacli.exe and save    │
│ XML output file for          │
│ processing.                  │
│ -read in ESP policy file used│
│ to specify required updates. │
│ -generate list of required and│
│ missing updates.             │
└─────────────────────────────┘
              │
┌─────────────────────────────┐
│     CHECK ANTVIRUS STATUS    │
│                             │
│ -write EICAR pattern file and│
│ check for AV removal.        │
│ -read 3rd party AV up-to-date│
│ value via WMI query.         │
└─────────────────────────────┘
              │
┌─────────────────────────────┐
│           FINISH            │
│                             │
│ -print or write cookie with  │
│ ESP status                   │
│ -open Internet Explorer at   │
│ selected user remediation    │
│ webpages.                    │
└─────────────────────────────┘
```

The utility includes the following features:

- Hard-coded checks for OS and service pack versions. For example, Windows XP Gold and SP1 are detected and cause a 'fail' condition.

- hooks to start Internet Explorer at remediation webpages: separate pages for critical update and AV status test fails.

- Can be custom integrated with other isolation systems via cookie reading capability.

- Configuration capability from central source. HTML form provided to edit configuration.

- Supports Windows 2000 (AV status check consists of EICAR check only), Windows XP, Windows Server and Vista.

- Installs in user temp folder, on completion, all files are deleted.

- No limitation on use of commercial AV products other than they support status access as in Windows Security Center.

- Provides log information to integrated server via cookie.

The utility mbsacli.exe works with the Windows Update Agent which runs on the computer and is also used by Microsoft Update. So both Microsoft Update and MBSA use the same source to gather update status. MBSA uses a database file called wsusscn2.cab downloadable from Microsoft to describe update information. In regard to licensing of MBSA, Microsoft specifies that all users accept the EULA. In order to use MBSA with ESP as described in this document, permission from Microsoft to allow one person from the institution to accept the EULA for all users is required.

See Appendix II for installation and configuration information.

User Remediation

An important design criteria of ESP is the need to have end users do as much of the repair or remediation of their computers as possible. The expectations of the end user are that they interact with Microsoft Update in order to install missing updates and with their AV product to ensure the AV is actively scanning files to be written to disk and signature update is functioning. When the user's system is missing critical security updates, the system directs users to Microsoft Update, which is Microsoft's resource for users to install updates. Remediation instruction and guidance is provided by topical web documentation – operating system update issues detected by ESP vulnerability detection cause an update web page to be presented to the user, antivirus issues cause an AV web page to be presented, etc.

Administering ESP

A NAC system using ESP can be implemented in the following two ways:

- Install and configure the entire ESP package which consists of the modified Netreg code and the ESP agent.

- Modify an existing NAC isolation environment by integrating the ESP agent.

Once installed, the ESP server administrator has the following tasks:

1. Ensure that site copies of the WSUS cab file are kept up to date. This is the file that is used by MBSA to determine what updates are defined. This task is usually performed after the monthly release of security updates by Microsoft.

2. Add the update identifiers to the ESP policy file. The admin has control over which updates ESP will be required to detect present. Normally, all updates in the current month would be added to this file but there may be reasons not to add all released updates – for example the discovery of a bug in a released update.

3. Configure the NAC system to require users to run the ESP agent. With the included Netreg system, this means that all registered user MAC addresses are deleted from the dhcpd.conf file. So users will be required to re-register and run the ESP agent when their current DHCP lease expires.

ESP Operations Experience

ESP has been in use for over two years and the following are two general observations surmised from IT staff who assisted with user remediation needs. Also, see Appendix IV for a list of specific issues experienced.

1. Unusual ESP failures and error messages are often a result of the machine under test being compromised (virus/worm/trojan) in some way.

2. It has been noticed that users who run ESP for the first time often have trouble or are unfamiliar with running the Microsoft Update system and require introductory training from the local Help Desk.

3. ESP administrators can arbitrarily specify when users are required to run the ESP agent. In practice, it has been observed that some residence administrators will trigger an ESP check within days of the Microsoft monthly update release (the delay is to ensure the updates are functional), others will trigger updates less often. The minimum trigger rate is at the beginning of the fall and winter academic terms. The campus wireless network administrator will trigger ESP checks at the beginning of the fall and winter terms. In both cases, regardless of when the update is issued, if the update addresses a critical vulnerability which has the potential to be exploited and cause possible degradation of the institution's network, then an ESP trigger is immediately configured.

4. For users to run ESP in the Windows OS environment, they must do so as a user with administrative rights on the computer. The tool was designed this way because the remediation function, MicrosoftUpdate, requires those rights to install updates. So when ESP is added to a network access environment in which both managed (user may not have admin rights to the computer) and unmanaged computer connect, it will be necessary to distinguish between these groups at network access time. At the University of Toronto, this is done locally during network authentication and authorization time – users who are classified as undergrads are required to run the ESP check, users who are classified as faculty and staff are not required to run the test.

Appendix I – Installation and Configuration of ESP server

The ESP server software has been tested on Linux Fedora Core 2, Core 4 and Core 5 (with some minor changes as placement of DHCP server files). Disable SElinux.

These packages have to be installed before installing ESP server software: dhcp, squid, apache, mod_ssl, subversion. These Perl modules Text::BasicTemplate, Net::Netmask, Net::IMAP::Simple, Mail::POP3Client, Net::LDAP, Test::More. Install perl-CPAN via yum.

After you downloaded and uncompressed the ESP server files into a folder you can start installing and configuring it by using the script init.sh.

Using init.sh

Init.sh will completely configure a dedicated UNIX platform to operate as an ESP server. The following is a step-by-step explanation:

It is best to run init.sh from a console rather than a network connection due to firewall changes. Two IP addresses have to be configured on a single interface – which init.sh will do. But the first time init.sh is run, NetworkManager must be disabled:

```
>service NetworkManager stop
```

```
>chkconfig NetworkManager off
```

Run the init.sh script.

```
>chkconfig NetworkManager on
```

```
>service NetworkManager start
```

Make sure that for however many times you run init.sh, it completes at least once. DO NOT make httpd, squid, or any other config edits until after init.sh completes successfully. The script will overwrite manual edits – so keep this in mind.

The administrator is prompted for permission to install configuration files in a standard location, then is prompted for specific configuration parameters that are substituted into configuration templates. Here is an explanation of each required parameter:

1. Apache user/group (required) added to /etc/passwd and /etc/group.

2. Server network configuration REAL IP: routable IP address. A DNS name will be looked up for this address – if there isn't one, a prompt will be issued for a name.

3. Server network configurationREAL netmask: routable subnet mask in the form of octet.octet.octet.octet.

4. Server network configuration PRIVATE IP: non-routable IP address.

5. Server network configuration PRIVATE netmask: non-routable subnet mask in the form of octet.octet.octet.octet.

6. Netreg REAL subnet (routable range used by Netreg to issue DHCP IPs)

7. Netreg PRIVATE subnet (non-routable range used by Netreg to issue DHCP IPs)

8. Routable gateway IP.

9. Domain name: top level domain name, eg. utoronto.ca

10. DNS server IP:

11. MAC address blocking feature – bandwidth limiting packet rate (Default:) Use default value on setup.

12. MAC address blocking feature – bandwidth limiting burst rate (Default:) Use default value on setup.

13. DHCP REAL pool range (can be smaller than REAL subnet size to allow for static IP range) : beginning IP address<space>ending IP address. Eg. 208.64.128.12 208.64.12.254

14. DHCP PRIVATE pool range (can be smaller than PRIVATE subnet size to allow for static IP range) : beginning IP address<space>ending IP address. Eg. 10.10.10.12 10.10.10.254

15. HTTPD_USER and HTTPD_GROUP: added to httpd.conf and used to set file permissions.

16. HTTPD_SERVER_ADMIN_EMAIL: to httpd.conf

17. HTTPD_SERVERNAME: If hostname is entered, it must be resolvable. IP address is acceptable.

18. HTTPD_ACCESS_CONTROL_LIST: in the format required by httpd.conf eg. 127.0.0.1 10.10.10.0/24 192.168.1.0/24 localhost

19. User prompting complete.

Notes:

a) init.sh - './init.sh –A' overwrites old files. This script may need to be run a number of times in this way due to the need to install missing perl modules.

b) start.pl - copies files from local repository to needed places

c) config.pl - configurator for local box

d) In /etc/sudoers, comment out:

```
Defaults requiretty
```

Post-Installation

- run serviceconf to disabled unnessesary services and auto-start the needed ones, like httpd,squid

Add this to the root's crontab

```
0-59/1 * * * * /usr/local/bin/refresh-dhcpdconf
```

Make sure that the ESP server clock is syncronized with a NTP server

```
20 2 * * * /usr/bin/rdate -s <put_ntp-server_ip_here>
```

Dhcpd.leases file can be saved for future references. To backup them to /var/lib/dhcp/hist folder add this line to crontab:

```
28 17 * * * cp /var/lib/dhcp/dhcpd.leases
/var/lib/dhcp/hist/dhcpd.leases.`/bin/date \+\%F\-\%T`
```

If you are planning to use Bandwidth limiting functionality add this to crontab after you installed netregbandw

```
0-59/1 * * * * /sbin/iptables-save > /etc/sysconfig/iptables
0-59/1 * * * * /etc/init.d/netregbandw save
```

Configuring an ESP Server to Work with an ESP Agent

The ESP Agent (checker_nr.exe) gets its real-time configuration from an ESP server. Configuration parameters are set in /etc/netreg/esp.cfg

They are configurable via http://<ESPSERVER>/cgi-bin/admin/config.cgi

Below is the screenshort of the configuration screen:

In the ResNet ESP environment /cgi-bin/admin/ folder is protected with a password.

Here are the parameters:

| Key-Value	Description

ESPpassURL=	the URL of the 'pass' page, can be left blank
email=my@email.com	email for notification about static IP addresses, can be left blank
savetimenum=2	number of min/hours/days/weeks/years to save Username/Room in a cookie
savetimetype=months	save registration Username/Room for'savetimenum' min/hours/days/weeks/years in a cookie
staticIPnotify=yes	message will be sent to 'email' if static IP monitoring scripts [1] are running and staticIPnotify is set to 'yes'
avcheck=enabled	enables AntiVirus checking
avchecklog=enabled	enables AV loging to a central server where checker_nr.exe resides
avuptodatecheck=enabled	enables AV up-to-date checking
eicarcheck=enabled	enables EICAR test
eicarcheckonly=disabled	Rely on EICAR test only not on presence of any AV
pwdcheck=disabled	Not available
dhcpvendorcheck=enabled	This will check if a Windows host sends a Windows-specific option in a DHCP request to avoid forged User Agent info. dhcpd.conf has to have 'set my-vendor-class = option vendor-class-identifier;'
nextpage=	Replaces /cgi-bin/next.cgi

Appendix II – Installation and Confiuration of ESP Agent

Building the Agent

To build the ESP agent executable from source:

1. install the latest NSIS SDK which is available from http://nsis.sourceforge.net.

2. Install NSIS XML plugin v1.6 by Shengalts Aleksander. This file is included in the ESP distribution but it needs to be copied to the <NSIS_home>/Plugins folder. This file is essential – newer versions may not work.

3. Create a home folder for ESP agent files.

4. Edit two configuration files in the ESP agent home folder: checker_cfg.txt and checker_cfg_nr.txt. Edit the policy_file_site key and enter the URL of the local site where ESP centrally managed files are located.

5. To compile the ESP agent source file (checker.nsi), right-click the filename and select NSIS Compile from the menu.

6. The resultant executable is named checker.exe. This is the stand-alone version of the agent. When run on a computer, it checks the update and AV status and prints a report in a text window (see below).
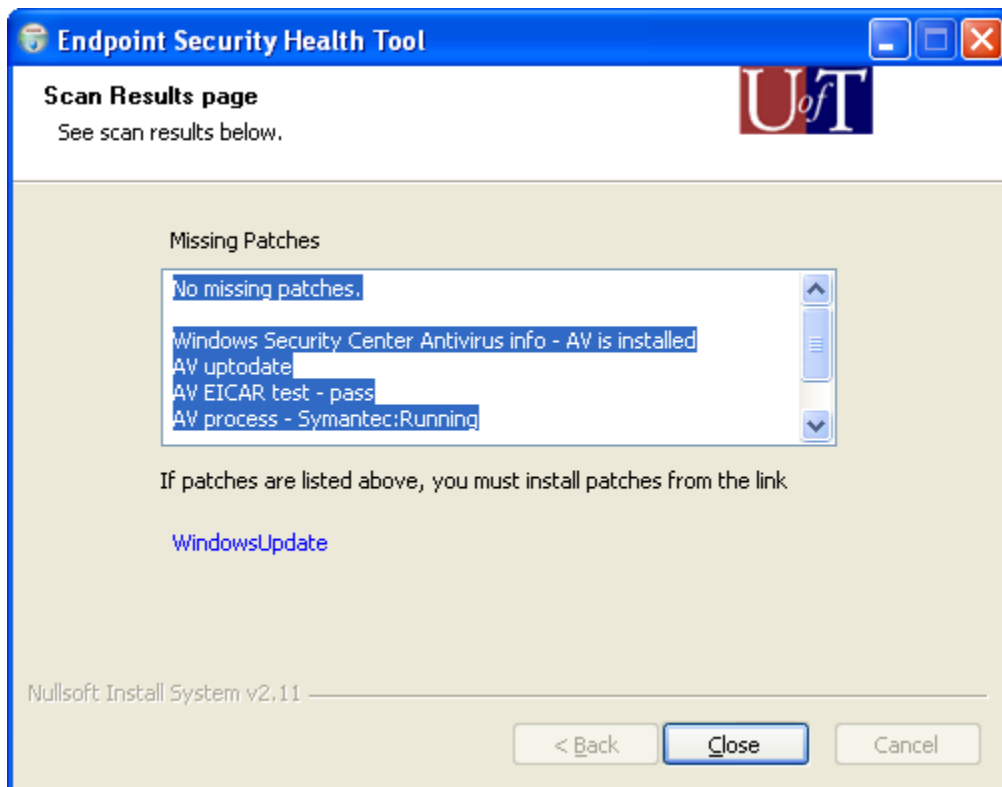
To create an agent that integrates with the ESP server or other NAC system, rename the file to: checker_nr.exe. To be specific, if the filename of the agent contains an underscore, it will read the checker_cfg_nr.txt file to configure itself. Otherwise, it reads the checker_cfg.txt file.

The NSIS source code file, checker.nsi, can be perused for customization of labeling, colours and images.

Configuring the Agent

See **Appendix I – Configuring an ESP Server to work with an ESP Agent** for details on configuring the agent remotely. The file ESP_update_policy.txt contains a list of the Microsoft updates that are required to be installed on a user's computer, one per line. The agent reads the file each time it is executed and uses the information to determine what updates are required. The following example illustrates this: if mbsacli.exe detects that MS07-050 is missing, and MS07-050 is listed in ESP_update_policy.txt, then the utility will return a fail. If MS07-050 is not listed in ESP_update_policy.txt, then the utility will return a pass. This file must be updated whenever new updates are provided by Microsoft (see Administering ESP).

Also, the file wsusscn2.cab must be downloaded from Microsoft and make available for local download. To summarize, the URL which specifies the central files required by checker should contain the following files:

Filename	Description
checker.exe	The stand-alone executable version of the ESP agent.
checker_nr.exe	The ESP agent which integrates with the ESP server – it outputs status in an HTTP cookie.
ESP_update_policy.txt	The text file that contains the ESP administrator-specified Microsoft updates that will be required on the tested computer.
wsusscn2.cab	The MBSA offline scan file which is downloadable from microsoft.com.
log.php	File used by checker to pass log information to the ESP server and/or the web server serving the central config files.
version.txt	Contains the current version number of the agent. It is used by the agent to detect version mismatches which may occur if a user tries to run an agent stored on the local machine.
avfail.htm, avfaq.htm	HTML files called by checker in the event of a failed AV check.
WindowsUpdateAgent20-x86.exe	Required for some older versions of Windows.

Appendix III – Utilities

Scripts to check for static IP addresses

mac-detect.sh **-** reports on active MAC addresses that are not registered in the ESP dhcpd.conf file. These may be the result of a user bypassing DHCP and assigning a hard-coded IP.

This is an example of what should be placed in to the crontab.

```
0,15,30,45 * * * * /usr/local/bin/mac-detect.sh
10,25,40,55 * * * * /usr/local/bin/staticIPsummary.pl /var/log/mac-rpt.txt
/var/log/mac-rpt.summary
15 * * * * /usr/local/bin/staticIPcheck.sh
```

Blocking MAC address

It's possible to block a host with a particular MAC address from accessing the outside world. There are two types of blocking. Please note that in either case the host has an unregistered (10.x.x.x) ip address obtained from the DHCP server running on the NetReg box.

Quarantine (or Partial) blocking

The host has access to the same resources (Microsoft patches,Symantec,CNS site) as a regular unregistered client, but it cannot get a registered ip address until administrator unblock it.

That is controled via /var/lib/dhcp/netreg_ether.deny file.

The checking is done in cgi-bin/register.cgi just before writing to the dhcpd.conf.new

The "netreg_ether.deny" file entry format is:

MAC_ADDRESS<space>Some text to display for a user

Blank lines and lines started with # are ignored.

example:

00:0c:29:5d:7c:d7 This host is blocked because of the Gaobot worm infection.

00:0c:29:c0:ec:1d The host is scanning others for LSASS vulnerability.

Full Blocking

In the case of full blocking host can only access the NetReg webserver and hosts on the local network. Every attempt to go to the outside world brings the user to the status page on the NetReg box telling him that the MAC address has been blocked and the reason for the blocking (controlled via an entry in /var/lib/dhcp/netreg_ether.deny file).

To enabling MAC blocking save the original /var/www/htdocs/index.html

16

under name /var/www/htdocs/index_orig.html - it will be read and displayed by /cgi-bin/status.cgi if the user MAC is not blocked.

Copy ./netreg/htdocs/index_redir.html to /var/www/htdocs/index.html

This file will redirect everyone to /cgi-bin/status.cgi .

Create a blank /var/lib/dhcp/netreg_ether.deny file.

Copy ./netreg/cgi-bin/status.cgi to /var/www/cgi-bin/ if it's not there.

To fully block a MAC address click on in /cgi-bin/admin/admin.cgi

for a partial block click on or you can do it manually via /usr/local/bin/netreg_block_mac.sh.

Bandwidth Limiting

It is possible to limit bandwith for a particular registered IP address.

It is done via a DHCP default gateway option and an IPTable firewall rule.

We have to add runlevel information for system services

to run /etc/init.d/netregbandw at boot time.

Thi should be added in to the crontab to save bandwidth rules between reboots:

0-59/1 * * * * /etc/init.d/netregbandw save

To impose bandwidth limit to a specific IP click on in /cgi-bin/admin/admin.cgi

MAC-IP address history

To save history information about MAC-IP address mapping and times add this to crontab:

28 17 * * * cp /var/lib/dhcp/dhcpd.leases
/var/lib/dhcp/hist/dhcpd.leases.`/bin/date \+\%F\-\%T`

The search interface is avaible via this script /cgi-bin/admin/hist.cgi

Appendix IV – ESP Help Desk Event Summary

Description	Resolution
Checker.exe reports failed AV check – also incorrect AV type for correct AV install. (F06)	WMI holds details of uninstalled AV which need to be removed manually. Run webmtest: command: wbemtest namespace to connect: root/SecurityCenter query: SELECT * FROM AntiVirusProduct
Checker.exe fails during mbsacli.exe check – error message: An error occurred while scanning for security updates (0x8024402c). (F06)	User laptop configured for updates from 3rd party (IBM).
Checker.exe fails during mbsacli.exe check – can't get local machine name due to an error:1060 (F06)	User computer not running 'Client for Microsoft Networks'
User can't complete the registration cycle for CWN. (F06)	Re-register MAC address.
Checker.exe reports missing patch MS07-042 on Vista while Windows Update does not. Probably because a trial version of MS Office Home and Student 2007 expired and Microsoft does not provide updates for it. (F07)	Manually install MS07-042 for Office. Workaround: Removed MS07-042 from the central ESP configuration file temporarily. MS07-042 still be installed for OS (but not for Office) with other patches.
AV is reported to be not up-to-date. Cannot update Symantec AV. (F07)	Might be because Windows Live OneCare interferes with it.
Vista with Windows OneCare trial fails EICAR. (F07)	Modified checker to detect up-to-date version of OC and set EICAR status to true.
Checker.exe reports it's an old version. Problem could be ESP inability to download version.txt file from the central server. That was observed on a Vista machine that had LiveUpdate left after Norton AV and had Symantec AV installed (F07)	Uninstalling Symantec and LiveUpdate and re-installing Symantec back has solved the problem. Additional comment: remember to turn off third party firewalls during ESP check
Error message: Windows Update client is missing. Mbsacli.exe reports "illegal system dll relocation" related to shell32.dll.	Check for presence of BricoPak Vista Inspirant by CrystalXP. There is a bug and MS Advisory wrt MS07-017 – the recommended solution is to remove Vista Inspirant.

Appendix V - Fedora Core 15 Changes

HTTPD

The init.sh script does not configure the httpd correctly as is. Please use the following to set up the file /etc/httpd/conf/httpd.conf:

The init.sh script will copy the new, unconfigured httpd.conf file to httpd.conf.ORIG. Make sure the following configuration is entered in this file:

```
User apache
Group apache

ServerName <fully qualified domain name of the ESP server>

<Directory />
    …
    Options FollowSymLinks
    AllowOverride None
    Order deny,allow
    deny from all
    allow from 127.0.0.1 10.10.xxx.0/23 142.151.xxx.0/23  localhost
    …
</Directory>

<Directory /var/www/htdocs>
    …
    Options Indexes FollowSymLinks MultiViews
    AllowOverride None
    Order deny,allow
    deny from all
    allow from 127.0.0.1 10.10.xxx.0/23 142.151.xxx.0/23  localhost
    …
</Directory>

<Directory /var/www/icons>
    …
    Options Indexes MultiViews
    AllowOverride None
    Order deny,allow
    deny from all
    allow from 127.0.0.1 10.10.xxx.0/23 142.151.xxx.0/23  localhost
    …
</Directory>

<Directory /var/www/cgi-bin>
    …
    Options None
    AllowOverride AuthConfig
    Order deny,allow
    deny from all
    allow from 127.0.0.1 10.10.xxx.0/23 142.151.xxx.0/23  localhost
    …
</Directory>

ScriptAlias /Squid/cgi-bin/ "/usr/lib/squid/"
```

```
<Location /Squid/cgi-bin/cachemgr.cgi>
    order allow,deny
    allow from 127.0.0.1 localhost
</Location>

<Directory /var/www/cgi-bin/admin>
    AuthType Basic
    AuthName "ESP Server Admin Page"
    AuthUserFile /etc/httpd/htpasswd
    require user espadmin
</Directory>
```

Run the command `httpasswd` to add the user espadmin and password.

Run the following to configure httpd to start at boot time:

```
chkconfig –levels 35 httpd on
```

Squid

Comment out the following config items in /etc/squid/squid.conf

```
# httpd_accel_host
# httpd_accel_port
# httpd_accel_with_proxy
# httpd_accel_uses_host_header
```

Add the following line:

```
http_port 3128 transparent
```

Run the following to configure squid to start at boot time:

```
chkconfig –levels 35 squid on
```

DHCPD

In the file /etc/dhcpd/dhcp.conf.new:

Replace: `ddns-update-style ad-hoc;`

With: `ddns-update-style none`

To start dhcpd at boot time:

```
systemctl enable dhcpd.service
```

Edit the file /etc/sysconfig/dhcpd:

```
DHCPDARGS="-cf /etc/dhcpd/dhcpd.conf -lf /var/lib/dhcp/dhcpd.leases -q"
```

IPTABLES Firewall

The init.sh script enters the value 'eth0' for the network interface. If this is not the name of the network interface, you will need to edit the config file:

```
/etc/sysconfig/iptables
```

Restart the firewall using:

```
/etc/init.d/iptables restart
```