Cybersoft.com

CyberSoft White Papers

Secrets of the VFind Security Tool Kit Professional Plus

Version 1.00 February 2000

by Peter V. Radatti

radatti@cyber.com
www.cybersoft.com

Pre-Preamble - May 1, 2000

The "Secrets" white paper has been on our web site for a couple of months and it appears to be fulfilling my expectations except in one area that was neglected. That area is an explanation of what the different packaged forms of the VFind Security Tool Kit are. This Pre-Preamble should correct that matter. Basically there are four packaged versions of VFind. The first package is the VSTK (VFind Security Tool Kit). The second package is the VSTKP (VFind Security Tool Kit Professional). The third package is the VSTKCW (VFind Security Tool Kit for Cyber War). The final package is VSTKSE (VFind Security Tool Kit Special Edition). VSTK is a subset of VSTKP which is a subset of VSTKCW. VSTKSE is basically a special version of VSTK/P (VSTK or VSTKP) that includes system integration services at a fixed fee. Here is a table that explains what is contained in the three main VFind packages.

TOOL	VSTK	VSTKP	VSTKCW	Available Separately
VFind	Y	Y	Y	N
MvFilter	Y	Y	Y	Y
CIT	Y	Y	Y	Y (CIT/THD ComboPak)
UAD	Y	Y	Y	Y

THD	Y	Y	Y	Y (CIT/THD ComboPak)
Bhead	Y	Y	Y	N
JDIS	Y	Y	Y	N
VGUI	S	S	Y	N
Robotmode	S	S	Y	N
Avatar	N	Y	Y	Y
NTI	N	N	Y	Y
NTI-CRYPTO	N	N	R	R
Unix Wrappers	N	N	Y	Y
RMI	N	N	Y	Y (Windows NT Only)

Y - Yes Included, N - Not Available Here, S - Available Soon, R - Restricted Sales

Preamble

One of the things that has aggravated me is the fact that our customers are not extracting the full value from the VFind Security Tool Kit (VSTK). Typically, the customer will learn the one or two tools they want when they purchase the product and ignore the remainder. This generally means that the customer is learning the VFind and UAD tools while ignoring CIT, THD and everything else. This is a problem because we created each of these tools to solve a different part of the overall computer security issue. If you are not using all of the basic tools provided then you are not maximizing your protection or investment and your system may be at risk.

We attempted to solve this problem with training classes but many of our customers don't have the training budget or time to attend a class. Many of these customers will call CyberSoft asking for configuration advise. Usually, we can communicate the important features and configuration options of the tools that the customer overlooked. This is usually a real eye opener for the customer and several then surprise us by finding new, unsuspected, uses for our products. In fact, one governmental customer wrote us a letter telling us how the CIT tool saved their system after a hacker break-in. They originally thought they were just buying a antivirus scanner, not an entire security toolkit. Surprise! Here, now, is my attempt to put some of this telephone consultation knowledge into a white paper for easy distribution and digestion. No training budget required!

There are only three ways to secure computers. The first is compartmentalization. Compartmentalization basically means that the system is isolated and locked up. Everything else is a compromise with various strong and weak points. The second method is called fortressing. This process attempts to make a fort or castle out of the computer by hardening the security. It keeps the bad guys out. Unfortunately modern computers are network connected and are very complex. Even with the best fortressing technology known someone, somewhere will penetrate the fortress and gain entrance. The third method is reactive security. Reactive security assumes that unauthorized people are going to get in or authorized people are going to make an unauthorized modification to the

baseline. It then deals with that issue. ***The most important part of reactive security is baseline configuration control.*** In other words, keep the system baseline where it belongs, keep the system under control and keep it performing its function as long as possible in a hostile environment.

Each of the three methods have drawbacks. The best overall solution is to combine all three. This is a common solution sometimes called a ring security system. The outermost ring provides compartmentalization to the extent possible. On a network this may be provided by configuration of the system, router and firewall. The second ring is fortressing. Remove all known exploits from the system and harden the configuration to make it as hard as possible to gain entry or control while allowing enough functionality to allow the system to continue performing its functions. The inner most ring is the reactive ring. Maintain the baseline configuration at all costs. Correct unauthorized changes to the baseline. Remove all attacks that gain entry. Automate the process so that discipline is maintained.

CYBERSOFT TOOLS AND THEIR USE IN THE TERTIARY SECURITY RING MODEL

	Available for:				
Tools	Compartment (Lock it up)	Fortressing (Harden it)	Reactive (Baseline Control)	Unix	Win-NT/ 2000
VFind	yes	yes	yes	yes	yes
CIT	no	no	yes	yes	yes
THD	no	no	yes	yes	yes
UAD	yes	no	yes	yes	yes
NTI	yes	yes	no	yes	yes
NTI-Crypto	yes	yes	no	yes	yes
RMI	yes	yes	no	moot	yes
Wrappers	yes	yes	no	yes	moot
MvFilter	yes	no	yes	yes	yes
Avatar	no	yes	yes	yes	yes
Robotmode	no	yes	yes	yes	yes

VFIND

VFind is the virus scanner and pattern analysis tool. It is unlike any other virus scanner in existence. It was the first antivirus scanner for Unix, the first heterogeneous virus scanner and the first scanner to incorporate a full virus description language, CVDL. Unlike most virus scanners, it actually searches for attacks in a file based upon what the file actually is. Most virus scanners assume that the filename is a description of the file type. VFind determines the file type by direct examination of the file's

contents. This makes VFind significantly more powerful than a virus scanner that only searches in files with the ".com" and ".exe" filename extensions for Microsoft executable viruses because it is not reliant upon a filename which in a hostile environment (such as a virus attack) could be wrong. Without this additional functionality, a mere filename change can be used as a form of stealth attack. In addition, this allows VFind to examine data in a byte stream in which filenames may not exist. This can be a significant feature if your computer is network attached.

VFind is also heterogeneous. This is critical in a server environment. Often a server will contain executable programs for network attached client systems of a different architecture. A very common example of this is a Unix server providing network disk services to Microsoft NT workstations. In this case, the Unix system can harbor viruses for the NT system even though it is itself immune to that attack. VFind solves this problem by simultaneously searching for Unix, Microsoft (MSDOS, boot sector, Win-32 and OLE Macro), Macintosh, Amiga and Java viruses.

VFind also includes the CVDL system. This system allows the user to define new attacks or any other type of information that can be examined by advanced pattern analysis. One such use for the CVDL system is to search for words or phrases that are not allowed on a system. These phrases could be proscribed as part of a organization policy such as sexual harassment or it could be part of a compartmentalization policy for handling classified information or for restricting what programs may reside on a computer by direct examination. It also allows for reactive processing of espionage attacks in which data is being moved within a system. Finally, CVDL allows for very fast updates of the VFind tool to search for new attacks without the need for replacement of the binary executable.

MVFILTER

The MvFilter program disinfects OLE documents (Microsoft Word, Excel and PowerPoint) from macro viruses (both VBA and Word Basic). It does this in the same way that all antivirus programs disinfect macro viruses, by removal of the macro. The difference is that MvFilter was designed as a tool. As such it can be used for compartmentalization purposes in addition to it's reactive disinfection role. As a compartmentalization tool, MvFilter can be used to pro actively prevent all macro virus infections, including new unknown infections, by automatically stripping all macros from OLE documents as they enter a system. This is a reasonable policy since it provides 100%, infallible protection from the fastest grown virus threat in the world in exchange for losing the ability to use macros. Since very few people use macros this should not be troublesome to them. Users who need macros can use authorized methods to allow the macros they need while still preventing damage from unauthorized macros.

A second use for MvFilter is as part of a document warehouse archival and management system such as Documentum. MvFilter has successfully interfaced and operated with these systems. Generally, these systems manage hundreds of thousands to millions of critical documents. MvFilter automates the process of document baseline control by providing a consistent format, free of all macros, in addition to preventing attacks.

CIT

We have recently modified the CIT process to dramatically accelerate the speed of the process.

CIT is a fantastic tool that complements Avatar. It has multiple uses and can tighten baseline configuration control down to a single bit or be used with surgical precision on an entire system or single file. The CIT tool produces a database of cryptographic hash values for every file it is directed to manage. In normal operation, this is every file on the system. Once the database is produced, it will report on every file that has been modified, added or removed. It can also be utilized to locate duplicate files or specifically locate any known file by the files hash signature. Additionally, it produces a target opportunity list of all files that have been added or modified on the system for virus scanning by VFind. When utilized with the LBH (Loop Back Head) and LBT (Loop Back Tail), which provides a feedback loop for infected files, it allows rapid virus scanning of a system without the danger of missing content. Files which are known to be clean and have not changed do not need to be scanned a second time.

CIT is also the best Baseline Configuration Control (BC2) tool. It allows the security officer to zero in on BC2 problems. Since all changes to a system are tracked, it becomes a trivial effort to detect baseline configuration problems. This is especially true if a hacker breaks into a system or an authorized user makes unauthorized changes. It can even detect changes caused by hardware degradation or outside electronic forces such as magnetic pulses or radiation. One more benefit of using CIT as a BC2 tool is that it becomes possible to diagnose and correct configuration problems with computers in a substantially shorter period of time than would be possible without the tool.

Another use for CIT is personnel monitoring. A trained security officer can determine a great deal about the user activity on a system by examining CIT reports. It is usually possible to tell if a user is not doing their job or is attempting to tamper with the system baseline configuration. CIT reported changes in system log files reveal which process have been executed and does so in a known time frame. It is not necessary to utilize the highly malleable system timestamp on files, you know that the detected activity took place in the window between CIT executions. This is in addition to identifying saved email, edited files, database files that were modified, insuring that files that must not exist (such as known Trojan horses or obsolete programs/documents) don't and generally determining what is happening and not happening in the system.

Other uses for CIT include insuring that the contents of a tape, disk or email attachment are what they are suppose to be. By referencing a secure database of known CIT hash values it is possible to determine if the contents have degraded or modified. This is extremely valuable when dealing with programs or critical documents distributed over a network or archived in off line storage. It is a 100% reliable way of knowing what was sent was what was received and what you thought was on removable media, is in fact, what is on the media. The CIT MD5 hash values are one way traps. It is not possible to determine the contents of a file by knowing the hash value. This means that it is possible to securely distribute hash values over open channels for even the most sensitive of documents.

The output from CIT is highly formatted which allows it to be read by data reduction programs. This means that CIT reports can be collected from very large networks into pools at central locations where data reduction techniques can produce an aggregate report. Aggregate reports of this type can reveal stealth attacks, including very hard to detect slow stealth attacks over a large number of systems. This report can be used as a quick way to determine if a very large number of network attached computers remain true to their baseline configuration. When operating in a hostile environment, this means that all of the computer equipment can be verified from one report as being correctly configured and operational. Systems which do not conform to the baseline configuration can be easily identified. Depending upon which options are selected, the central pool of CIT reports can also be utilized to locate a single file on one computer in a network of hundreds of thousands of computers.

UAD

The UAD tool solves two difficult problems, identification and decomposition. Decomposition of a file to it's smallest indivisible parts (universal atomic disintegration using classical Greek language meanings) is a difficult problem. First the program must have infallible identification of the file in order to decompose it. This is not a problem for UAD which identifies the file by direct examination of it's contents. Most decomposition tools assume the contents of a file by it's filename. If the file is named "xyz.zip" the decomposition tool will assume that the file is a "zip" compresses composite file. UAD does not make any assumptions. This also allows UAD to identify data in a byte stream where filename information may not exist. This is important in a network environment.

Secondly, decomposition is critical to proper pattern analysis. There is no value in virus scanning a compressed, composite or encoded file since the encapsulating technology will hide the contents from examination. This is why UAD is able to decompose email, including attachments in uuencode and mime formats. It is also able to decompose tar, gnu gzip, pkzip, zip2exe, Unix compress and other formats. UAD will continue decomposition recursively until every part of the file has been decomposed into a state that is known to be a terminus (atomic state) or has been decomposed into an unknown format. Most unknown formats are already in the atomic state or are moot.

A benefit of the UAD system besides it's uses for virus scanning is its ability to decompose many formats of encapsulated file. This can save a lot of time when the file format is not directly compatible with the system on which it resides. The user just executes UAD with the file he wishes to decompose and UAD performs the rest. Unfortunately, many times when a user uses a tool other than UAD to decompose a file into its parts the tool will place the decomposed files in multiple places on the system. UAD solves this problem by forcing the current working directory to be the top level directory for the purposes of decomposition. This allows a user or system administrator to have full control over the installation of a new program without "splating" programs and data all over the system in an uncontrolled way.

AVATAR

(note: equations are formatted for Netscape 3.0 & above. Other browsers may not view them correctly)

Avatar maintains the system Baseline Configuration. It does so by executing system security policies that act as an intrusion detection and response system. The most important function of Avatar is, response. If the system Baseline Configuration is modified, for any reason, it will be detected by Avatar and returned to the correct Baseline Configuration. The value of Avatar's response system is that it enforces discipline by non-subjective automated process which can execute many times per day.

Intrusion is defined as any unauthorized modification to the system Baseline Configuration. The reason for this broader than normal definition is that it allows for unauthorized modifications by authorized and unauthorized personnel. When an unauthorized person breaks into a computer their actions will always be dictated by their goals. If they are a passive reader then their activity will be captured in the system logs. If they are using the system as a platform for further attacks then they will download attack programs for execution and they will want to insure future access. To insure future access they will have to change the Baseline Configuration. Modification of the system logs such as changing permissions, insertion of Trojan back doors into critical system applications, modifications of the Baseline in any form or just plain destruction of critical system files can all be detected and corrected by Avatar. The addition of new inappropriate files to a system can be detected by CIT.

The ability to maintain the Baseline Configuration also provides extensive immunity to new unknown software attacks within the Baseline. If a binary or script virus infects a file then the file will be overwritten by the Baseline version of the file. This effectively destroys the virus and is far superior to any form of virus disinfection used by any other company. When a virus infects a file, it modifies it. In the process of infecting the file, it is common for the file to be damaged. [$f_V(a) = a'$] The disinfection process used by most antivirus companies may or may not remove the actual virus. It is most common to not remove the virus but merely change program pointers so that the program executes around the virus without executing the virus. If necessary, the virus is then modified so that it is no longer detected by the same antivirus program as a live virus. This preserves the damage created by the virus and potentially adds new damage if the pointers are modified incorrectly. [$f_d(a') = a''$, $a \neq a''$] In addition, not all viruses, especially new unknown viruses, can be disinfected. None of these problems exist with Avatar since a captured copy of the original file is used to overwrite the infected file. [$f_{avatar}(a') \circ a$] This also works for all forms of software attacks in Baseline configured programs, not just viruses or hacker attacks.

Avatar security policies can be maintained on a file by file basis or on an entire system. Security policies that can be maintained are:

(e - EXISTENCE) The existence rule states that the file(s) must exist. It does not infer any other rules. This is extremely valuable for files that must exist but whose contents constantly change. Two examples of files of this type are log files and password shadow files.

(p - PERMISSIONS) The permissions rule states that the permissions of a file must not deviate from the baseline configured permissions. Generally all system command, log and configuration files have critical permission settings that must not change. This can be combined with the "e" and "o" rules to provide maximum protection for files whose contents need to change over time.

(o - OWNERSHIP) The ownership rule states that the owner of the file must not deviate from the baseline configured ownership. For example, the ownership of the password shadow file can be used as a back door into a system , while ownership of a log file can be used to stealthily erase evidence of activity on a system.

(s- CRYPTOGRAPHIC SIGNATURE) The cryptographic signature also known as a hash value states that an alarm should be activated if the contents of the file change but no further action should take place. This is extremely useful when used within other programs and when attempting to catch hackers. An additional use is to allow a script or program to verify that a critical file conforms to the baseline configured contents without overwriting any deviations from the baseline.

(c - FILE CONTENTS) The contents rule states that the contents of a file must not change. If the contents change for any reason, the file is overwritten with the correct baseline configured contents.

A rule exists (!) to force a pathname to not be baselined. This is most valuable for scratch file areas such as the Unix "/tmp" and "/var/spool" areas. The opposite also exists (R - RECURSIVELY) which forces Avatar to baseline the entire path recursively. Finally a shorthand rule (E - EVERYTHING) exists which has the same operation as selecting the "eposcR" rules.

The Avatar Database was designed so that it can be read only. This means that the Avatar Check, Avatar Correct programs along with an Avatar Database can all exist on a CD-ROM or other read only format, (NFS, DVD, Zip, etc). A read only version of Avatar can not be hacked. Once invoked by a background process such as Unix Cron or remotely invoked from a Security Server the Avatar Correct system will automatically detect and correct any problems.

A significant feature is that an alternative Avatar Database can be defined for automatic fail-over if the primary Avatar Database has a failure. For example, if the primary Avatar Database is located on a network disk and the network failed, Avatar can continue processing with a local database. In fact, multiple Avatar Databases can exist and be invoked in any order necessary. This allows Avatar to be used for multiple purposes including rapid system software update distribution and configuration for thousands of workstations without the need for personnel to visit them. System customization can also be accomplished by a smaller Avatar database which can be stored locally or at a central location.

One of the advantages of the Avatar system for Internet based systems such as web servers is that shortly after a hacker modifies a web page, the Avatar system can be automatically invoked and restore the damaged page. If Avatar is run dozens of times per day then the hacker would literally have to break in and modify the system dozens of times per day. This is a huge incentive to leave the system alone.

THD

THD answers the question of how do you find a chameleon Trojan horse attack when there is no contents to scan. The chameleon Trojan horse attack works because a user is able to redirect a system command to a program of the same name in a different location. The chameleon may or may not have contents. If the chameleon has contents then it can be located using VFind. If it does not have contents then THD can locate it because by definition the filename must have the same name as another program on the system or the attack will not work.

BHEAD

BHead is a simple tool. It is a program that reads a specified number of bytes from a file or byte stream and writes them to standard output. While this is not a complex task the problems the tool can solve can be complex. One such problem is that Unix systems running on Intel based PC architecture hardware can be infected with boot sector viruses. Unix systems do not have a convenient way to read just the boot sector for scanning. Using the Unix "dd" command the entire raw disk drive can be read and output as a byte stream. Scanning the entire drive would be a redundant waste of time, however, using Bhead the byte stream can be cut to just the portion of the drive that contains the boot sector. Bhead can also operate on files, floppies and tapes.

JDIS

JDIS is a tool that performs fast disassembly of Java byte code. There is a version of JDIS called JADE (Java Advanced Disassembly analysis Engine) built into VFind. It is important to disassemble Java byte code prior to virus scanning because it is the only way to associate constant pool entries with opcodes. This association allows the virus scanner to scan for what is actually going on in the program instead of guessing. VFind is the only antivirus tool to provide scanning of Java disassembly.

JDIS is also provided as a standalone tool so that a security officer can disassemble a Java byte code program into disassembly for manual analysis. This can be valuable when confronted with a potential new, unknown Trojan Horse written in Java.

VGUI

This is the graphical user interface for all of the VSTK tools. The tools can optionally continue to be utilized from the command line and as components in script files. The GUI includes the ability to configure all of the tools, execute all of the tools, disable tools and reset/create databases. It includes status lights for all of the major databases and provides a visual indicator of tools that are "active".

The most significant advantage of VGUI is that it captures the factory settings for all of the tools including the manufacturer's best practices. This means that a user who does not have the time to read the documentation and understand the tools can still gain as much benefit as an expert. This "captured knowledge" is the default value for all tools utilized via VGUI. On the other hand, VGUI

provides a safety net for the novice user because no matter how damaged a customization becomes the VGUI provides a method to reset everything back to the factory settings and start over.

Finally, the learning curve for graphical enabled tools is significantly less than for command line tools. This will encourage the user to utilize more tools in ways that solve specific problems.

ROBOTMODE

This is the default mode for VGUI. It is also a standalone program that can be executed by the operating system's clock function, by hand, or remotely by an authorized process. Robotmode examines the entire VSTK security suite of tools and determines which tools need configuration. If the user has already configured a tool then no further configuration will be made by Robotmode. The program then configures any tools that still require configuration, creates any databases that has not already been created and then executes all of the security tools in the most logical sequence possible.

If a CIT database has not been created then Robotmode will create a CIT database for the entire system while simultaneously virus scanning the system using UAD and VFind with Loop Back. If an Avatar configuration file has not been created then it will create one followed by executing Avatar Create to create the Avatar database. It will also run THD and all of the other security tools as predetermined to be appropriate by CyberSoft.

The second time Robotmode runs all of the tools will have already been configured and the databases built, either by Robotmode or by the user. At this point, Robotmode will run CIT, VFind, UAD, LBH with LBT, THD and Avatar Check/Correct. If necessary it will also run MvFilter to remove any macro viruses.

Robotmode will continue to expand over its life cycle. It is the first attempt to force discipline into a process that was previously inconsistent by means of automation. Robotmode encapsulates best practices for the use of the VSTK tools.

NTI

The Network Traffic Interceptor is a new concept in virus scanning network traffic. Instead of using proxies like other products, the NTI program actually implements an antivirus firewall in the computer itself. This is a new standard in virus scanning because it intercepts any potential threat prior to it's access to user application memory.

This is very important for the following reasons.

1. 80% of all attacks originate within the inside LAN. A single point of control firewall can only stop attacks from the outside LAN. NTI implements a firewall for antivirus purposes on every workstation, solving this problem.
2. If a large number of people all attempt to use a centralized firewall then it will suffer bandwidth throughput problems. By locating the most process and bandwidth intensive part of the firewall

on the local system there is no aggregate problem.

3. [***Most Important***] By intercepting viruses at the IP packet layer, they can be analyzed prior to reaching an application where they can potentially execute. This is especially true of email and web browsers with extensions such as Java.
4. In addition, NTI with VFind allows end users to default Java and Java Script to on in their Web Browsers while insuring that known Java attacks are blocked, no matter what the source. This can be important when using smart web sites that utilize Java and Java Script. Without this ability, the user has to choose between not accessing those sites, turning Java on and off on a per site basis or accepting the risk of leaving it on all of the time.

NTI is capable of scanning any traffic going into or out of a system. In NTI version 20, the first commercial version, CyberSoft has pre configured NTI to virus scan for all ftp file transfers, http downloads, SSL secure sockets layer, https downloads, pop3 email, smtp email and SMIME email attachments (See NTI-CRYPTO below for SSL, https and SMIME information.)

NTI can be executed on both the local workstation and on servers. Currently, March 2000, NTI is available for Solaris 2.5.1 and above, HPUX 11.0 and above and Microsoft Windows NT 4.0 with Service Pack 5.

NTI-CRYPTO

The NTI-CRYPTO option of the NTI program allows for effective virus scanning of encrypted data. Unless data is decrypted prior to scanning, any viruses that may exist will be hidden by the encryption algorithm along with the intentional hiding of data. NTI-CRYPTO deals with SSL encryption and SMIME encryption.

SSL encryption is used for all encrypted web page transactions (https) and is the backbone of e-commerce. NTI-CRYPTO uses the NTI system to intercept all encrypted SSL traffic prior to user access then performs real time cracking and scanning of the SSL encrypted connection. If the data contains viruses then the connection is broken and the user is sent a message as to the nature of the problem. The virus never enters the system application layer.

NTI-CRYPTO also intercepts and virus scans all SMIME encrypted email messages and attachments. It does this by reading the SMIME decryption keys from the Netscape Browser SMIME key database. It then uses these keys to decrypt the message for scanning. If the message or it's attachment is infected with a virus then the message is blocked prior to entering the system application layer.

The NTI-CRYPTO technology is one of a kind. CyberSoft invented (patent pending) this sole source technology. No other computer security company has any product that can scan encrypted data until after it has entered the system and has been decrypted at the application layer. Once a virus is at the application layer the system is threatened.

One final note on the NTI-CRYPTO product is that it can be used instead of cryptographic key escrow as a method of legal wire tap for the purpose of detecting and stopping espionage.

[March 2000 - Currently the NTI-CRYPTO product is only available to the United States Federal Government. It will be released for commercial sale, in the United States, in the forth quarter of 2000.]

UNIX WRAPPERS

The Unix Wrappers tools are programs that take the place of and envelope the standard Unix tar, cpio and mount commands. The tar and cpio programs are common ways of reading data into a Unix system from removable media such as tape or floppy diskette. The mount command is how file systems are mounted onto the Unix filesystem for access. This includes network drives supplied by NFS (Network File System) and CD-ROM drives. When any of the wrapped tools are used to import files into the protected Unix system the files are automatically virus scanned by VFind. If the files are uninfected they are allowed to enter. If they are infected or contain proscribed data they are blocked. This is a proactive tool that prevents infection.

WINDOWS REMOVABLE MEDIA INTERCEPTOR

The Windows Removable Media Interceptor (RMI) intercepts all removable media mounts such as floppy diskette, CD-ROM and zip disks prior to user access for virus scanning. If the media is infected then the user is denied access. RMI prevents infected files from entering a system.

This site certified 508 Compliant