Cloud-based Digital Forensics Evaluation Test (D-FET) Platform

Prof William J Buchanan, Richard J Macfarlane, Flavien Flandrin, Dr Jamie Graves, Bill Buchanan (Dell Computers), Dr Lu Fan, Dr Elias Ekonomou, Niladri Bose, Robert Ludwiniak

Centre for Distributed Computing and Security,
Edinburgh Napier University
{w.buchanan,r.macfarlane,f.flandrin,j.graves,l.fan,n.bose,e.ekonomou,r.ludwiniak}@napier.ac.uk,William_Buchanan@Dell.com
http://cdcs.napier.ac.uk

Abstract. This paper outlines the specification of the Cloud-based D-FET platform which is used to evaluate the performance of digital forensics tools, which aim to detect the presence of trails of evidence, such as for the presence of illicit images and determination of user accounts from a host. Along with measuring key quality metrics, such as true-positives, and false-positives, it also measures operational performance, such as for the speed of success, CPU utilization and memory usage. This is used to determine the basic footprint of the package-under-test. The paper presents a proof-of-concept of the system using the VMware vSphere Hypervisor (ESXi) within the vCenter Cloud management infrastructure, which provides a cluster environment, and supports the creation and instantiation of a well-defined virtual test operation system. The infrastructure has been used within a teaching environment for two semesters, and has been shown to cope well in terms of performance and administration. Two key evaluation points related to whether a cloud-based infrastructure will provide improvement on existing stand-alone and workstation-based virtualisation are related to the improvement in energy consumption and in the CPU utilization footprint for each virtual machine. Thus the results show some metrics related to the energy and CPU consumptions of the created digital forensics instances, which can be used to justify the improvements in energy consumption, as opposed to stand-alone instances, and in the scalability of the infrastructure.

1 Introduction

There are a wide range of digital forensics tools on the market, including both open and closed source systems. Unfortunately, there is not a standardized range of tools which allow for the *quality* of the evaluation tools to be assessed. The two main methods of creating an evaluator is either to create a host simulator, which provides known responses to requests from tools, or to create an emulated environment, which creates an actual working version of an operating system and/or disk images. Simulators are fairly easy to create, but suffer in that they

can never actual respond in the same way as a real-life operating system. Thus the emulated version is often the best choice. Along with this the emulated version can be created with a well-known script which pre-prepares the host with known activities, which match to the range of test vectors that are to be run.

The aim of the Cloud-based D-FET platform, an innovation based on collaborations with the Home Office, is to produce an evaluation platform which evaluates the quality of digital forensic test tools. This will then to be used evaluate tools, and thus allows software vendors to get feedback on how they can improved their software. A key element is that the infrastructure provides for an automated assessment of the quality of system using pre-prepared virtual instances which are setup from an interaction script. This then supports a range of evaluation, including using well-know scripts, for validation tests, to full random testing for a long-term evaluation. The infrastructure presented in this paper has been used within a teaching environment to teach advanced methods and digital forensics and security, and has coped well. Thus a fully developed infrastructure would thus also have great benefits within the teaching of undergraduate and postgraduate students, especially in creating instances which continually vary, so that no two students will get the same virtual instance to test against. Along with this, the system could be used within an assessment infrastructure in the same way that evaluators are used for Cisco device configuration [10].

2 Literature Review

Digital Forensics is used to produce evidence which can be used in legal proceedings, thus the *quality* of this evidence is vital [25]. Meyers et al highlights the need for improved standardisation of the digital forensic methods used to provide evidence, thus mitigating challenges of such procedures in court. They argue that other disciplines make use of such standards, such as for the National Institute of Standard and Technologies (NIST) who have developed standards for forensics DNA testing [27] which is kept updated to take into account new technologies. In 2007 at least 14 different investigation methodologies existed, however, none of these cover the entire range of digital forensics procedures [34].

2.1 Certifications and Tools

Beebe states that "Digital forensics largely lacks standardization and process" [7]. Along with this, certifications are also becoming available for computer forensic examiners, such as the Certified Computer Examiner (CCE) [23] and the GIAC Certified Forensics Analyst [22]. These can help experts to choose the appropriate tools to use when performing investigations. Tools currently used to carry out digital forensic procedures can be either open source such as the Sleuth Kit [12] and PyFlag [16], or proprietary such as EnCase (Guidance Software Inc) [35] and FTK (AccessData Corp) [17]. The majority of these tools, though, are created in isolation and based on independent research, and very few tools are

based on previous research or built to any common standards [21]. For example, Law enforcement is currently using around 150 different automated tools to perform digital investigation [33], which offers users an extensive amount of functionality, moreover, new versions are published regularly. It is thus not conceivable to evaluate all features; therefore it is essential to optimise the evaluation process [37].

Another problem for forensic tool evaluation has been exposed by [13]; in terms of the licensing or tools. Results produced by tools should be assessed, therefore the methodology used by the tool have to be analysed. Commercial tool vendors generally will not release source code for their software, but open source tools offer the possibility to assess the quality of their internal components and flows. A key point raised by the Carrier is that vendors of closed source tools should, at the very least, provide an overview of the design specifications, so that the testing of the tools features can be carried out more effectively.

There is currently little in the way of robust assessment and comparison of these digital forensics tools. Results from some industry standard tools such as EnCase, are simply accepted by the computer forensic community [25]. This can be based on the reputation of the vendor, and not always due to the testing tools against any particular standard.

Software packages have been known to be purchased based on nothing more than web-based reports [25]. Therefore, researchers argue that it is vital for digital forensics to develop an evaluation methodology for tools that will be accepted by vendors, practitioners, and the scientific community [21] [26]. Failure to provide quality assessments of the tools used, could impact on the credibility of digital forensics practitioners, along with the credibility of the evidence produced. For instance, [24] proved that the *dd* tool do not always copy all sectors from a hard disk, and proved that this error did not come from *dd* but than from the Linux kernel (versions up to, and including, 2.4). Without testing and evaluation, such problems may not have been found and corrected.

Tentative work to create a framework for evaluating digital forensics tools has been carried out [3]; however, the current status of the framework is not known. The Scientific Working Group on Digital Evidence (SWGDE) produced guidelines to validate tools [36], but no actual validation of existing tools has been performed to date. A key organisation which could lead a major initiative is NIST, who have set up a working group on Computer Forensics Tool Testing (CFTT), which aims to provide a set of specifications to assess digital forensics tools [28]. The CFTT project focuses on disk acquisition; using both hardware and software write blocking. Specifications have been developed for both, and they are currently still in draft format. Testing has been carried out on several tools, and the results produced so far are available. More recently, the project created a set of specifications for forensics media preparation tools [32], and also for the forensic analysis procedures of string searching [30] and deleted file recovery [31], although a full test methodology for analysis tools has yet to be created [12]. The main goal of the CFTT project is to provide measures, to ensure that the tools used during an investigation produce correct and consistent

results. This gives the assurance for experts that the evidence recovered by these tools will be acceptable in court. The CFTT project, though, does not go as far as comparing tools, or rating them for specific procedures, and they only recently created specifications forensic image analysis features, which are still in a draft version. At the present time the test data being used is currently unavailable. There is thus an issue related to reproducing the tests to assess the validity of the findings within digital forensic tools.

2.2 Forensic Evidence Data Sets

Digital Forensics should be recognised as a branch of Forensics Science, therefore, scientific concepts for experiments have to be applied. A scientific experiment should be controlled and reproducible, and experiments have to be documented, to permit peers to reproduce them under the same conditions and be able to confirm the results. This reproducibility means researchers can build on the results of previous research. Garfinkel et al [19] exposed this issue, and argued that lack of standardised data sets, or corpora, that are available for research purposes is holding back research and professional development. Without the ability to use the same data set, it is not possible for researchers to perform the same experiments and reproduce their results. In their paper they present a taxonomy of current corpora, and define new forensic data sets for research use. This is extremely important, as it allows different groups to evaluate their methods and tools against the same set of data, and from a common baseline. When distributing corpora researchers also have to be careful to ensure that none of the distributed files infringe copyrights. For instance, the Honey Project and the Digital Forensics Research Workshop (DFRWS) do not distribute images that contain Microsoft applications because of copyright issues [19]. Researchers can use binary scrambling techniques which modify binaries, therefore associated programs will not be able to run, but the files are still available for analysis [20]. Using such techniques might solve the copyright issue in some cases but creates a new issue. As binaries have been modified, their hash signature will be modified. Thus, forensics techniques that use databases of known binaries to discard unimportant files, or to search for known evidence, will not behave correctly [15].

NIST has started the Computer Forensic Reference Data Sets (CFReDS) Project [29], which aims to provide simulated sets of digital evidence, which can be used to test forensic tools. Another limited set of corpora, to be used for tool testing, has been brought together by Brian Carrier [14]. Both projects have created corpora, and the CFReDS Project additionally provides some of the data files and resources used to create the data sets. [19] Garfinkel et al have also developed a set of corpora which are available to researchers. These corpora are divided in four categories: disk images; real disk images; real data corpus - only available with an IRB clearance; and a list of real, but unrestricted, file corpus. The latest is composed of one million files divided in 1000 different categories. Such corpus could be used to generate disk images with known content, and

using the huge amount of files, it would be possible to create many different cases to test a large range of forensics analysis procedures.

2.3 Virtualised Testing Environments and Metrics

The usage of baseline data provides an excellence way to assess tool functionality, and they can then be compare using various metrics. Scripting could be used to control and standardise the content of these corpora. The usage of forensic evaluation metrics is also an evolving field such as for [2] who define Common Vulnerability Scoring System (CVSS) metrics to define forensic quality measures. Ayers et al [6] introduces a range of metrics to measure the performance of forensic tools, including absolute and relative speed, accuracy, completeness, and auditability.

Another useful technology is virtualisation, as this enables the creation of virtual machines that simulate different types of hardware on a single computer system. These virtual machines then behave as if they were independent computer systems [8], either through a hypervisor with paravirtualisation (such as with KVM, Citrix, VMware and Xen) or through full virtualisation (such as with VMWare Workstation and VirtualBox).

In a virtual environment, hardware is simulated but there can be problems simulating all possible types of hardware. Sometimes when trying to boot a disk image from a physical machine the OS might not be able to handle the new hardware. Therefore, particular files have to be created to resolve this issue. At the end of this process data will be considerably modified, breaking the first of the Association of Chief Police Officers (ACPO) principle: "No action taken by law enforcement agencies or their agents should change data held on computer storage media which may subsequently be relied upon in court" [1]. Evidence found in this way would not be able to be used in court [8], Bem et al, nevertheless, exposed a new way to see virtualisation for computer forensics analysis. A disk image from the suspected machine could thus be mounted within a virtual machine; an analysis of the system could be performed without following forensics principles by a non-forensics expert. Then, a certified digital forensics expert could analyse the machine following forensics principle to confirm the previously found results, thus, the evidence is likely to be acceptable in court. This method permits to save time as multiple people could look for evidences without having to follow constraining forensics methodology.

2.4 Usage of Cloud and Virtualised Environments within teaching

The usage of Cloud and virtualised environments is now well defined such as Brueckner [9] who have developed the CYber DEfenSe Trainer (CYDEST) virtualized training platform. This uses an automated assessment systems for trainees through a Web interface. Buchanan et al [11] have used it within a VMWare vCenter infrastructure to teach a wide range of modules including for computer security and digital forensics. In terms of digital forensics investigations, cloud

and virtualised systems could contribute a great deal, especially in terms of snapshotting captured images. Dorn [18], though, that there is a strong requirement to understand how virtual images are created within virtualised environments, especially in how they store their files and manage the processes and ancillary files and associated artefacts. If the methodologies used in virtualised environment are to be used as evaluators, and also within evidence gathering, it is important to understand their operation. Arnes [4] highlights that it is important to improve the credibility of virtualised systems for digital forensics by presenting ViSe, which is a virtual security testbed, where the attacks are defined as event chains, which are then replayed into the testbed.

3 Forensic Testing Platform

This section outlines the existing development infrastructure and the proposed platform for D-FET.

3.1 Platform design

Figure 1 outlines the proposed infrastructure where the package-under-test (PUT) is loaded into a virtualised cluster, such as within VMWave vSphere or Open Nebula. The Host Forensic Operation Instance (HFOI) is taken from a Host Forensics Image (HFI) Library, which contains a wide range of predefined instances, and which are created from the original Operating System (OS) installation disks. This includes well-known Windows, Linux and MAC OS images. The Digital Forensics Instance Creator (DFIC) is then used to take an image from the library, instantiate it, and then run an activity script within the created instance. Any digital media can also be loaded, such as a JPEG image or an AVI movie, from the Forensics Media Library (FML).

This type of platform is proposed as it most clearly creates a well-known instance, which has a well-defined activity trace. Along with loading a defined instance from the library, a pre-defined disk can be loaded from the DI (Disk Instance) library. This also allows for well-defined disk images to be loaded into the emulated host image.

3.2 Activity Scripting

The DFIC is responsible for creating the instance, and running the script. As much as possible the script is easy to create and interpret, thus a simple scripting language has been created to support the interpretation of the script. For example to create a static instance:

```
INSTANCE LOAD [Image=WINDOWS2003]
MOUNT INSTANCE [Disk=STANDARDDISK] AS [Partition="c"]
ACTIVITY LOAD [Number=12] [Type=JPEG IMAGES; Class=DRUGS]
    INTO [Folder=USER FOLDER]
    AT [Period=1 MINUTE][Interval=INTERVAL]
    FOR [User=Fred]
```

which will create an instance based on WINDOWS2003 from the HFI Library, and then load 12 JPEG instances from the DRUG classifications of images, using the STANDARDDISK disk image. This type of activity creates the host with predefine activity, but which does not have a timeline of activity. To create a timeline of activity, events can then be created, such as for a user login, deleting a file, and then logging out:

```
INSTANCE LOAD [Image=WINDOWS2003]
MOUNT INSTANCE [Disk=STANDARDDISK] AS [Partition="c"]
ACTIVITY LOAD [Number=12] [Type=JPEG IMAGES; Class=DRUGS]
  INTO [Folder=USER FOLDER]
  AT [Period=1 MINUTE][Interval=INTERVAL]
  FOR [User=Fred]
ACTIVITY EVENT [Event=LOGIN; User=Fred]
ACTIVITY EVENT [Event=DELETEFILE; User=Fred; File=JPEF IMAGES]
ACTIVITY EVENT [Event=LOGOUT; User=Fred]
```

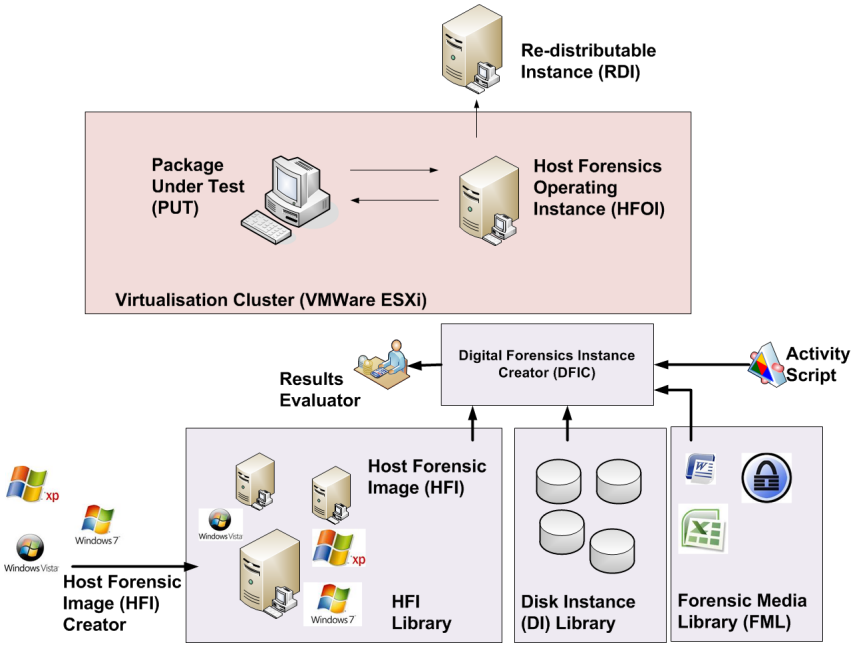


Figure 1: D-FET Platform

Once an instance has been created, it can then be cloned back into an instance library, which can be distributed as required. The robotic operation can be achieved with a wide range of tools. AutoIT [5] is automation software that can record user activities, and then play them back, thus recreating user actions on the system. It provides a high level interface for recreating activities within user space, thus simulating the high level functions called by a user. It can also be programmed with a bespoke, BASIC-like, scripting language to perform ac-

tivities within user space. These scripts interact directly with the Graphical User Interface (GUI). AutoIT is a fairly sophisticated piece of software, as it is able to interrogate the data behind GUI elements, which allows logic and decision flow to be built into the scripts if required.

3.3 Evaluation methods and metrics

The platform aims to create the required results from the running instance, and then uses two main quality measures:

- Performance. This is defined as the CPU utilization, memory footprint, and so on.
- Forensic Quality of Evaluation. This is a measure of false positives/negatives and true positives/negatives reported, along with the time to complete a challenge (such as the time taken by an investigator/student to find a certain activity or objects.

As a basic evaluator the Home Office have defined a range of evaluation tests. These are outlined in Table 1 (in the Appendix), where ValD defines the value of the detected value for the HUT, and ValS defines the actual number of detectable items on the HFOI. The measure of success is thus how well the tool finds the required activity.

There are a range of evaluation methods which can be used within D-FER, and these relate to the level of testing and evaluation required. These include:

- White box testing. When the script of activity is known and pre-prepared, and which can be used as a validation test by vendors.
- Gray box testing. This involves a well type of activity, but the actual detail of the evaluation is not known to the platform under test (PUT).
- Black-box testing. This involves a randomization of the activity based on the script.

3.4 Current Implementation

As a proof-of-concept VMware vSphere and the VMWare ESXi 4.1 hypervisor has been used to create a clustered environment, from which the instances can be created. Figure 2 shows the infrastructure, and which has been trailed within a teaching environment to a wide range of security and digital forensics modules over the past year [11]. The developed infrastructure has three main ESXi hosts (Socesx2, Socesx4 and Socesx3), and a main controller (Socesx1). The main controller runs: Lab Manager (which provides a Web browser interface which users connect too, to run their instances); a firewall/router (which allows certain types of traffic to be blocked, and routing between the private internal network and the external one); a shared data storage of 4TB (using iSCSI for fast access times); and vCenter (which is responsible for controlling the ESXi hosts). A key advantage of VMware vCenter is that virtual networks can be created which either connect to the external network (typically to the Internet), or can be completely

isolated from other networks and instances running within the infrastructure. Along with this instances can be fenced or unfenced, where particularly sensitive tools and content can be explored within a fenced environment which can have no contact with any other systems.

If the infrastructure is to be used to create a community cloud for digital forensics, and used in a teaching environment, a large shared storage is important as hundreds of instances need to be stored, and along with this a relevantly large memory is often required on the cluster hosts in order for them to run many instances at a time without extensive need for disk caching. While the controller does not have to be a particularly powerful computer, it is important that the clustered hosts can perform well, so the two main cluster servers (Socesx2 and Socesx3) were selected with the following specification:

Type: Dell PowerEdge R410

CPU: Intel Xeon 2.27GHz, 8CPUs (16 logical processors on two physical processors) Licence: vSphere 4 Advanced

Memory: 32GB

Each of the cluster hosts has two network connections, one which connects to an internal private network and the other to a router/firewall running on the controller. The internal network has been set for 192.168.x.x/16, which allows for more than 65,000 virtual hosts to be created, and which can be shared on the same network (this is important as it allows users to work together and use each other instance for security/forensics evaluations). The router on the controller then allows for external connections to the public network. For digital forensics applications this connection should be used only for transferring files or in downloading software. Figure 3 shows an example of a pre-prepared Windows 2003 instance, and a Linux tools is used to evaluate against it.

4 Evaluation

The vCenter infrastructure has been stressed tested over from Sept 2010 to March 2011 using student labs to evaluate its performance, and overall the infrastructure has coped well. In order to evaluate the limits of the system, a formal experiment measured the CPU utilization and power usage difference between launched VMs. Figure 4 illustrates the CPU utilization of the VMs using 100% loading. The power consumption was measured using a power meter connected to the power support of the server running the VMs. The footprint was thus measured as varying between 5W and 7W. Each VM shows a clearly defined CPU step-up of 15% or approximately 2GHz of CPU time. For eight running VMs the overall energy cost was 7.65W per VM. As the loading is at 100%, it is likely that most VMs would not be running at this maximum and there would be a considerable reduction in the overall CPU loading, and thus the system could cope with more VMs running at the same time. Figure 5 provides an example of the management console with the cluster nodes and the running instances. It should be noticed that most instances do not actually use the full CPU requirement for themselves.

In terms of the three cluster nodes in the vCentre infrastructure, the total CPU capacity is 45.272GHz, thus, running at 100% loading for all of the VMs would thus support a maximum of 22 VMs. This is the worst case scenario, and under testing over the year using students performing labs, it has been observed that the three nodes can support up to 35 VMs running Windows 2003.

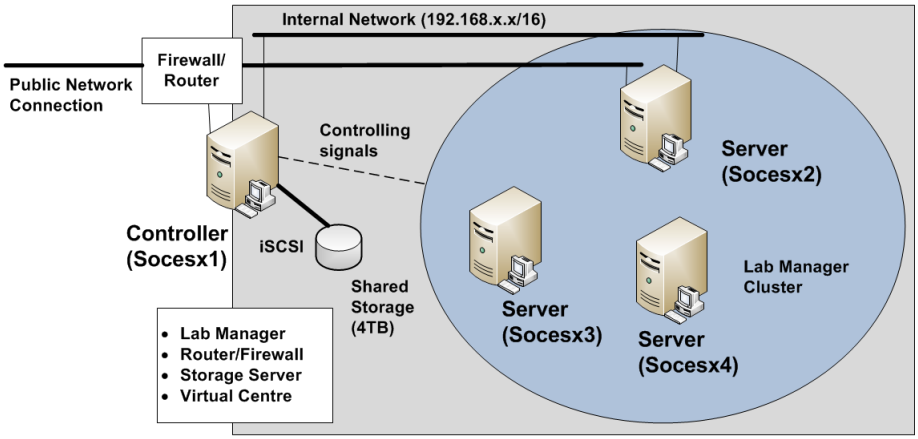


Figure 2: vCenter Infrastructure for virtualised digital forensics/security assessments

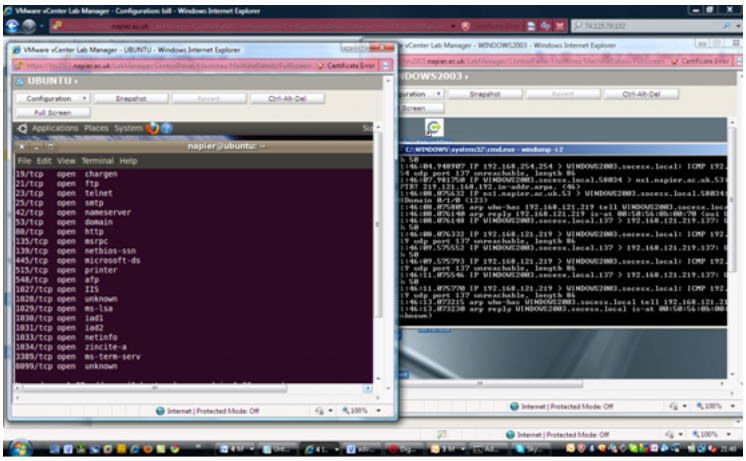


Figure 3: D-FET Proof-of-Concept using VMWare Lab Manager

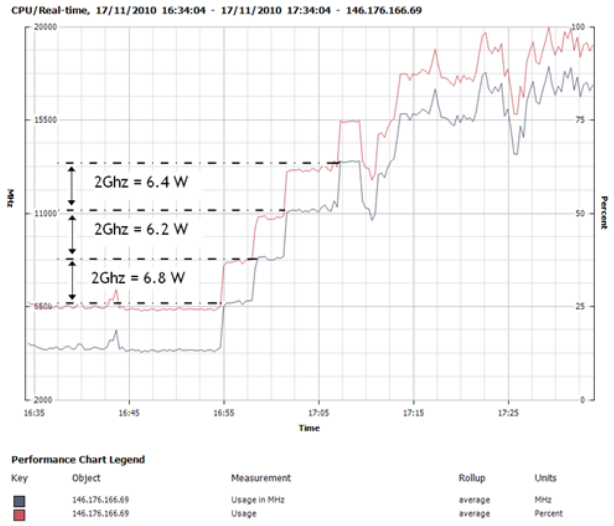
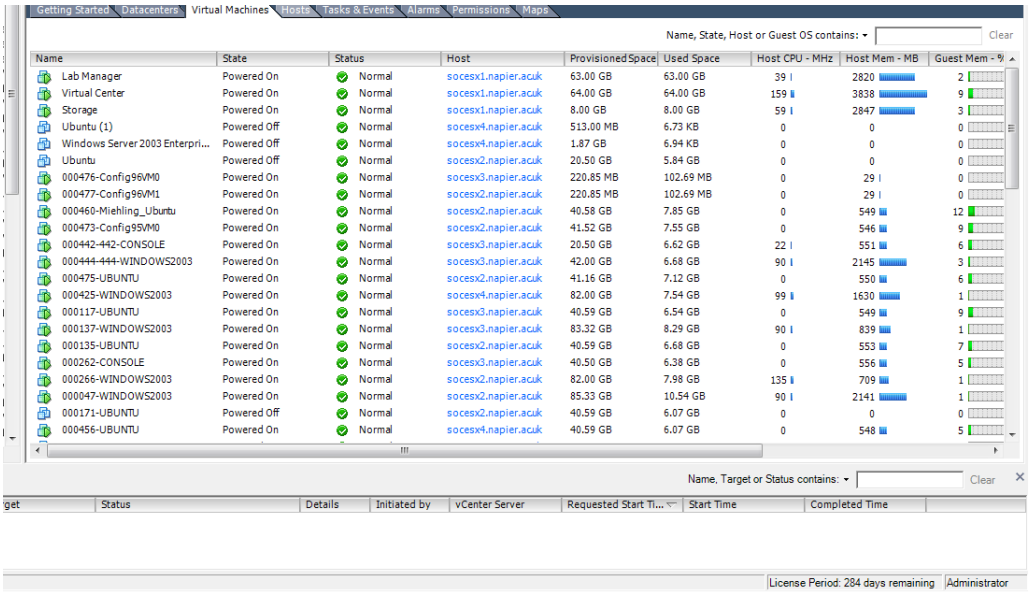


Figure 4: Virtualisation footprint



Name	State	Status	Host	Provisioned Space	Used Space	Host CPU - MHz	Host Mem - MB	Guest Mem - %
Lab Manager	Powered On	Normal	socesx1.napier.acuk	63.00 GB	63.00 GB	39	2820	2
Virtual Center	Powered On	Normal	socesx1.napier.acuk	64.00 GB	64.00 GB	159	3838	9
Storage	Powered On	Normal	socesx1.napier.acuk	8.00 GB	8.00 GB	59	2847	3
Ubuntu (1)	Powered Off	Normal	socesx4.napier.acuk	513.00 MB	6.73 KB	0	0	0
Windows Server 2003 Enterpri...	Powered Off	Normal	socesx4.napier.acuk	1.87 GB	6.94 KB	0	0	0
Ubuntu	Powered Off	Normal	socesx2.napier.acuk	20.50 GB	5.84 GB	0	0	0
000476-Config96VM0	Powered On	Normal	socesx3.napier.acuk	220.85 MB	102.69 MB	0	29	0
000477-Config96VM1	Powered On	Normal	socesx2.napier.acuk	220.85 MB	102.69 MB	0	29	0
000460-Miehling_Ubuntu	Powered On	Normal	socesx2.napier.acuk	40.58 GB	7.85 GB	0	549	12
000473-Config95VM0	Powered On	Normal	socesx2.napier.acuk	41.52 GB	7.55 GB	0	546	9
000442-442-CONSOLE	Powered On	Normal	socesx3.napier.acuk	20.50 GB	6.62 GB	22	551	6
000444-444-WINDOWS2003	Powered On	Normal	socesx3.napier.acuk	42.00 GB	6.68 GB	90	2145	3
000475-UBUNTU	Powered On	Normal	socesx2.napier.acuk	41.16 GB	7.12 GB	0	550	6
000425-WINDOWS2003	Powered On	Normal	socesx4.napier.acuk	82.00 GB	7.54 GB	99	1630	1
000117-UBUNTU	Powered On	Normal	socesx3.napier.acuk	40.59 GB	6.54 GB	0	549	9
000137-WINDOWS2003	Powered On	Normal	socesx3.napier.acuk	83.32 GB	8.29 GB	90	839	1
000135-UBUNTU	Powered On	Normal	socesx2.napier.acuk	40.59 GB	6.68 GB	0	553	7
000262-CONSOLE	Powered On	Normal	socesx3.napier.acuk	40.50 GB	6.38 GB	0	556	5
000266-WINDOWS2003	Powered On	Normal	socesx2.napier.acuk	82.00 GB	7.98 GB	135	709	1
000047-WINDOWS2003	Powered On	Normal	socesx2.napier.acuk	85.33 GB	10.54 GB	90	2141	1
000171-UBUNTU	Powered Off	Normal	socesx2.napier.acuk	40.59 GB	6.07 GB	0	0	0
000456-UBUNTU	Powered On	Normal	socesx4.napier.acuk	40.59 GB	6.07 GB	0	548	5

Figure 5: vCenter infrastructure

5 Conclusions

The Cloud-based D-FET system provides a proof-of-concept system which allows for the creation of well-defined test images which can be used to evaluation the quality of digital forensic tools. The vCenter infrastructure supports the easy management of virtual instances of a digital forensics community cloud, with the rights of assess carefully defined. While the VMware infrastructure works well there is a licencing cost, thus open source Cloud systems such as Open Ubuntu may provide a less expensive infrastructure for the academic community. At present the research group are working on an Open Ubuntu equivalent, but it is not as easy to setup in the same that the vCenter infrastructure has been created. The methods used in proposing D-FET allow for a scaleable evironment which is forever changing, and thus will provide investigators and students the opportunity to training on environments which are ever changing, but still provide the same type of challenge. This will allow allow for the evaluation of basic metrics such as determining the number of true-positives, which can be used to evaluate the tools, and also in assessing student performance.

The paper has also shown that there is a considerable saving in power consumption, in terms of each VM used, as opposed to standard desktop equivalent. At full loading, the system could cope with up to eight instances running on each cluster at a given time. The more nodes that can be added to the infrastructure will thus support more instances, at a time, while supporting a much more robust environment. Energy efficent can also be achieved by moving VMs from one cluster node to another, when the loading is fairly light on a specific cluster node, and then this machine can be put into power saving mode. Overall, in terms of energy efficency, quality of instances, and creating ever-changing environments, the Cloud-based D-FET platform seems to provide the opportunity to create a digital forensics community cloud, where there would be no need to distribute instances on DVDs, and where it would be possible to carefully control access to the instances created.

6 Appendix

Table 1: Basic evaluation tests

Test rule	PUT	HFOI
Presence of known illicit images	ValD	ValS
Presence of known illicit movies	ValD	ValS
Evidence of accessing/viewing/uploading/downloading illicit materia	ValD	ValS
Evidence of moving/copying/burning/printing illicit material to other locations	ValD	ValS
Preview of media files	ValD	ValS
User accounts - number & names	ValD	ValS
Presence of filesharing software	ValD	ValS
Filesharing history vs known bad files	ValD	ValS
Presence of counter-forensics software	ValD	ValS
Internet browsing history	ValD	ValS
Internet - cookies	ValD	ValS
Recent documents/files	ValD	ValS
Word processor documents - contents	ValD	ValS
Spreadsheet documents - contents	ValD	ValS
Text documents - contents	ValD	ValS
Databases - contents	ValD	ValS
Email messages received - contents	ValD	ValS
Email messages sent - contents	ValD	ValS
Email messages drafted - contents	ValD	ValS
Email contacts	ValD	ValS
Chatlogs - transcript	ValD	ValS
Chatlogs - attributed author transcript	ValD	ValS
Chatlogs - timed transcript	ValD	ValS
Open network connections - presence	ValD	ValS
Open network connections - ID details	ValD	ValS
Connected device history	ValD	ValS
Running processes/programs	ValD	ValS
Times/timelines of file access/modify	ValD	ValS
Timeline of general user activity	ValD	ValS
Typed search data histories	ValD	ValS
Stored passwords	ValD	ValS
Presence of encryption software	ValD	ValS
Presence of encrypted files/volumes	ValD	ValS
Presence of counter-forensics software	ValD	ValS
Presence of obscured files - signature/extension mismatch	ValD	ValS
Hidden files (unallocated space) - recovery	ValD	ValS
Deleted files - recovery	ValD	ValS
String searches for ASCII strings		
String searches for UNICODE strings		

References

1. ACPO. Good practice guide for computer-based electronic evidence. `http://www.7safe.com/electronic_evidence/ACPO_guidelines_computer_evidence.pdf`, 2003.
2. A.R. Amran, R.C.-W. Phan, and D.J.t Parish. Metrics for network forensics conviction evidence. *International Conference for Internet Technology and Secured Transactions. ICITST 2009*, pages 1 – 8, 2009.
3. C Armstrong. Developing a framework for evaluating computer forensics tools. Canberra,, 2003. Evaluation in Crime and Justice: Trends and Methods Conference, Australian Institute of Criminology.
4. Andr Arnes, Paul Haas, Giovanni Vigna, and Richard Kemmerer. Using a virtual security testbed for digital forensic reconstruction. *Journal in Computer Virology*, 2:275–289, 2007.
5. AutoIT. Autoit. `http://www.autoitscript.com/site/autoit`, 2011.
6. Daniel Ayers. A second generation computer forensic analysis system. *Digital Investion*, 6:34–42, 2009.
7. N Beebe. Digital forensics research: the good, the bad, and the unaddressed. In *Fifth annual IFIP WG 11.9 international conference on digital forensics*, January 2009.
8. D Bem and E Huebner. Computer forensic analysis in a virtual environment. *International journal of digital evidence*, 2007.
9. Stephen Brueckner, David Guaspari, Frank Adelstein, and Joseph Weeks. Automated computer forensics training in a virtualized environment. *Digital Investigation*, 5(Supplement 1):S105 – S111, 2008. The Proceedings of the Eighth Annual DFRWS Conference.
10. W. Buchanan. Correlation between academic and skills-based tests in computer networks4. *British Journal of Educational Technology*, 37:69–78, 2006.
11. William J Buchanan, Jamie Graves, Niladri Bose, Richard Macfarlane, Brian Davison, Richard Ludwiniak, and Bill Buchanan. Performance and student perception evaluation of cloud-based virtualised security and digital forensics lab. *HEA ICS Conference*, 2011. HEA ICS Conference.
12. Carrier. *File System Forensic Analysis*. Addison-Wesley Professional, Reading,, 2005.
13. B Carrier. Open source digital forensics tools. Technical report, @Stake, 2003.
14. Brian Carrier. Digital forensics tool testing images. `http://dftt.sourceforge.net`, 2010.
15. M Cohen and B Schatz. Hash based disk imaging using aff4. *Digital Investigation*, 7:121–128, 2010.
16. MI Cohen. Pyflag-an advanced network forensic framework. *Digital Investigation*, pages 112–120, 2008.
17. Access Data. Forensic toolkit 3. `http://accessdata.com/products/forensic-investigation/ftk`, 1997.
18. Greg Dorn, Chris Marberry, Scott Conrad, and Philip Craiger. Analyzing the impact of a virtual machine on a host machine. In Gilbert Peterson and Sujeet Shenoi, editors, *Advances in Digital Forensics V*, volume 306 of *IFIP Advances in Information and Communication Technology*, pages 69–81. Springer Boston, 2009.
19. S Garfinkel, P Farrell, V Roussev, and G Dinolt. Bringing science to digital forensics with standardized forensic corpora. *Digital Investigation*, 6:2–11, 2009.

20. S L Garfinkel. Automating disk forensic processing with sleuthkit, xml and python. pages 73–84, Berkeley,, 2009. Proceedings of the fourth international IEEE workshop on systematic approaches to digital forensic engineering, IEEE.
21. Simson L Garfinkel. Digital forensics research: The next 10 years. *Digital Investigation*, 7:64–73, 2010.
22. GIAC. Giac certified forensic analyst (gcfa), 2000.
23. ISFCE. Cce certification, 2005.
24. J D Kornblum. The linux kernel and the forensic acquisition of hard disks with an odd number of sectors. *International Journal of Digital Evidence*, 3(2), 2004.
25. Matthew Meyers and Marc Rogers. Computer forensics: The need for standardization and certification. *International Journal of Digital Evidence*, 3(2), 2004.
26. S Mocas. Building theoretical underpinnings for digital forensics research. *Digit Invest*, 1:61–68, 2004.
27. NIST. Quality assurance standards for forensic dna testing laboratories. `http://www.cstl.nist.gov/strbase/QAS/Final-FBI-Director-Forensic-Standards.pdf`, 1998.
28. NIST. Computer forensics tool testing (cftt) project. `http://www.cftt.nist.gov/`, 2003.
29. NIST. The cfreds project. `http://www.cfreds.nist.gov/`, Dec 2008.
30. NIST. Forensic string searching tool requirements specification. `http://www.cftt.nist.gov/ss-req-sc-draft-v1_0.pdf`, 2008.
31. NIST. Active file identification & deleted file recovery specification. `http://www.cftt.nist.gov/DFR-req-1.1-pd-01.pdf`, 2009.
32. NIST. Forensic storage media preparation tool specification, 2009.
33. NIST. Computer forensics tool testing (cftt) project overview, 2011.
34. M M Pollitt. An ad hoc review of digital forensic models. Proceedings of the second international workshop on systematic approaches to digital forensic engineering, 2007.
35. Guidance Software. Encase forensic. `http://www.guidancesoftware.com/forensic.htm`, 1997.
36. SWGDE. Recommendations for validation testing, 2009.
37. Tom Wilsdon and Jill Slay. Validation of forensic computing software utilizing black box testing techniques. Perth Western Australia,, 2006. 4th Australian Digital Forensics Conference, Edith Cowan University.