Chapter

Data Collection Techniques for Forensic Investigation in Cloud

Thankaraja Raja Sree and Somasundaram Mary Saira Bhanu

Abstract

Internet plays a vital role in providing various services to people all over the world. Its usage has been increasing tremendously over the years. In order to provide services efficiently at a low cost, cloud computing has emerged as one of the prominent technologies. It provides on-demand services to the users by allocating virtual instances and software services, thereby reducing customer's operating cost. The availability of massive computation power and storage facilities at very low cost motivates a malicious individual or an attacker to launch attacks from machines either from inside or outside the cloud. This causes high resource consumption and also results in prolonged unavailability of cloud services. This chapter surveys the systematic analysis of the forensic process, challenges in cloud forensics, and in particular the data collection techniques in the cloud environment. Data collection techniques play a major role to identify the source of attacks by acquiring evidence from various sources such as cloud storage (Google Drive, Dropbox, and Microsoft SkyDrive), cloud log analysis, Web browser, and through physical evidence acquisition process.

Keywords: distributed denial of service attacks, digital forensics, network forensics, web forensics, cloud forensics, mobile forensics

1. Introduction

In today's world, users are highly dependent on the cyberspace to perform all day-to-day activities. With the widespread use of Internet technology, cloud computing plays a vital role by providing services to the users. Cloud computing services enable vendors (Amazon EC2, Google, etc.) to provide on-demand services (e.g., CPU, memory, network bandwidth, storage, applications, etc.) to the users by renting out physical machines at an hourly basis or by dynamically allocating virtual machine (VM) instances and software services [1–3]. Cloud computing moves application software and databases to large data centers, where the outsourcing of sensitive data and services is not trustworthy. This poses various security threats and attacks in the cloud. For instance, the attackers use employee login information to access the account remotely with the usage of cloud [4]. Besides attacking cloud infrastructure, adversaries can also use the cloud to launch an attack on other systems. For example, an adversary can rent hundreds of virtual machine (VM) instances to launch a distributed denial-of-service (DDoS) attack. A criminal can also keep secret files such as child pornography, terrorist documents, etc. in cloud storage to remain clean. To investigate such crimes involved in the cloud, investigators have to carry out forensic investigations in the cloud environment. This arises the need for cloud forensics, which is a subset of network forensics. Cloud forensics is an application of scientific principles, practices, and methods to reorganize the events through identification, collection, preservation, examination, and reporting of digital evidence [5]. Evidence can reside anywhere in the cloud and it is more complex to identify the traces located in the cloud server.

The advancement of new technologies, frameworks, and tools enables the investigator to identify the evidence from trusted third parties, that is, cloud service provider (CSP). There are numerous techniques in cloud forensics that arises on the basis of cloud service models and deployment models. In the Software as a Service (SaaS) and Platform as a Service (PaaS) models, the customer does not have any control of the hardware and they need to depend on CSP for collecting the evidence, whereas, in the case of Infrastructure as a Service (IaaS) model, customers can acquire the virtual machine (VM) image and logs.

The forensic examiner isolates the attacked system in the virtualized environment by segregating and protecting the information from a hard disk, RAM images, log files, etc. This evidence is analyzed based on the artifacts of the attack traces left by the attacker [6, 7]. The forensic investigator relies on finding a series of information such as where, why, when, by whom, what, and how attack has happened. This chapter details the challenges in cloud forensics and also details the data collection techniques in the cloud.

2. Types of forensics

The forensic process is initiated after the crime occurs as a post-incident activity. It follows a set of predefined steps to identify the source of evidence. It is categorized into five groups, namely digital forensics, network forensics, Web forensics, cloud forensics, and mobile forensics.

- **Digital forensics**: According to National Institute of Standards and Technology (NIST) standards, it is the application of science to the identification, collection, examination, and analysis of data while preserving the integrity of the information and maintaining a strict chain of custody for the data.

- **Network forensics**: It identifies and analyzes the evidence from the network. It retrieves information on which network ports are used to access the information.

- **Web forensics**: It identifies the evidence from the user history, temporary log files, registry, chat logs, session log, cookies, etc. as digital crimes occur on the client side with the help of Web browser.

- **Cloud forensics**: It is the application of digital forensics in the cloud and it is a subset of network forensics. It is harder to identify evidence in cloud infrastructure since the data are located in different geographical areas. Some examples of evidence sources are system log, application log, user authentication log, database log, etc.

- **Mobile forensics:** It is the branch of digital forensics that identifies evidence from mobile devices. The evidence is collected from the mobile device as call history, SMS, or from the memory.

2.1 Cloud forensic process flow

The cloud forensic process flow is shown in **Figure 1**, which is described as follows:

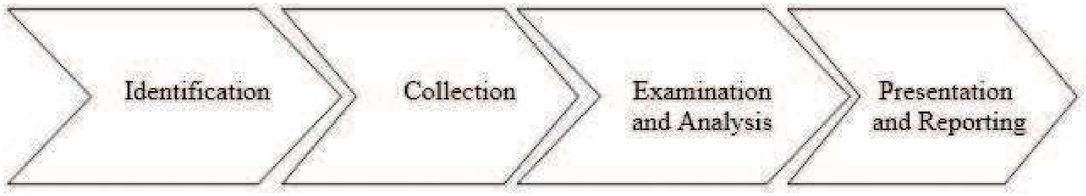


Figure 1.
Cloud forensic process flow.

- **Identification**: The investigator identifies whether crime has occurred or not.
- **Evidence collection**: The investigator identifies the evidence from the three different sources of cloud service model (SaaS, IaaS, and PaaS) [8]. The SaaS model monitors the VM information of each user by accessing the log files such as application log, access log, error log, authentication log, transaction log, data volume, etc. The IaaS monitors the system level logs, hypervisor logs, raw virtual machine files, unencrypted RAM snapshots, firewalls, network packets, storage logs, backups, etc. The PaaS model identifies the evidence from an application-specific log and accessed through API, patch, operating system exceptions, malware software warnings, etc.
- **Examination and analysis:** The analyst inspects the collected evidence and merges, correlates, and assimilates data to produce a reasoned conclusion. The analyst examines the evidence from physical as well as logical files where they reside.
- **Preservation:** The information is protected from tampering. The chain of custody has been maintained to preserve the log files since the information is located in a different geographical area.
- **Presentation and reporting:** An investigator makes an organized report to state his findings about the case.

3. Evidence collection

Evidence collection plays a vital role to identify and access the data from various sources in the cloud environment for forensic investigation. The evidence is no longer stored in a single physical host and their data are distributed across a different geographical area. So, if a crime occurs, it is very difficult to identify the evidence. The evidence is collected from various sources such as router, switches, server, hosts, VMs, browser artifacts, and through internal storage media such as hard disk, RAM images, physical memory, etc., which are under forensic investigation. Evidence is also collected through the analysis of log files, cloud storage data collection, Web browser artifacts, and physical memory analysis.

3.1 Cloud log analysis

Logging is considered as a security control which helps to identify the operational issues, incident violations, and fraudulent activities [9, 10]. Logging is mainly used to monitor the system and to investigate various kinds of malicious attacks. Cloud log analysis helps to identify the source of evidence generated from various devices such as the router, switches, server, and VM instances and from other internal components, namely hard disk, RAM images, physical memory, log files etc., at different time intervals. The information about different types of attacks is stored in various log files such as application logs, system logs, security logs, setup logs, network logs, Web server logs, audit logs, VM logs, etc., which are given as follows:

- *Application log* is created by the developers through inserting events in the program. Application logs assist system administrators to know about the situation of an application running on the server.

- *System log* contains the information regarding date and time of the log creation, type of messages such as debug, error, etc., system-generated messages related to the occurrence, and processes that are affected by the occurrence of an event.

- *Firewall log* provides information related to source routed packets, rejected IP addresses, outbound activities from internal servers, and unsuccessful logins.

- *Network log* contains detailed information related to different events that happened on the network. The events include recording malicious traffic, packet drops, bandwidth delays, etc. The network administrator monitors and troubleshoots daily activities by analyzing network logs for different intrusion attempts.

- *Web server log* records entries related to the Web pages running on the Web server. The entries contain history for a page request, client IP address, date and time, HTTP code, and bytes served for the request.

- *Audit log* records unauthorized access to the system or network in a sequential order. It assists security administrators to analyze malicious activities at the time of attack. The information in audit log files includes source and destination addresses, user login information, and timestamp.

- *VM log* records information specific to instances running on the VM, such as startup configuration, operations, and the time VM instance finishes its execution. It also records the number of instances running on VM, the execution time of each application, and application migration to assist CSP in finding malicious activities that happen during the attack.

Due to the increase in usage of network or new release of software in the cloud, there is an increase in the number of vulnerabilities or attacks in the cloud and these attacks are reflected in various log files. Application layer attacks are reflected in various logs, namely access log, network log, authentication log, etc., and also reflected in the various log file traces stored on Apache server. These logs are used for forensic examination to detect the application layer attacks. **Table 1** indicates the various attack information and the tools used for log analysis of different types of attacks. **Figure 2** shows the sample access log trace (**Table 2**).

- ***Sample Network Log Entry***

```
[**] [1:1407:9] SNMP trap udp [**] [Classification: Attempted Information
Leak] [Priority: 2] 03/12–15:14:09.082119 192.168.1.167:1052 - > 172.30.128.27:162
UDP TTL:118 TOS:0x0 ID:29101 IpLen:20 DgmLen:87.
```

- ***Sample Firewall Log Entry***

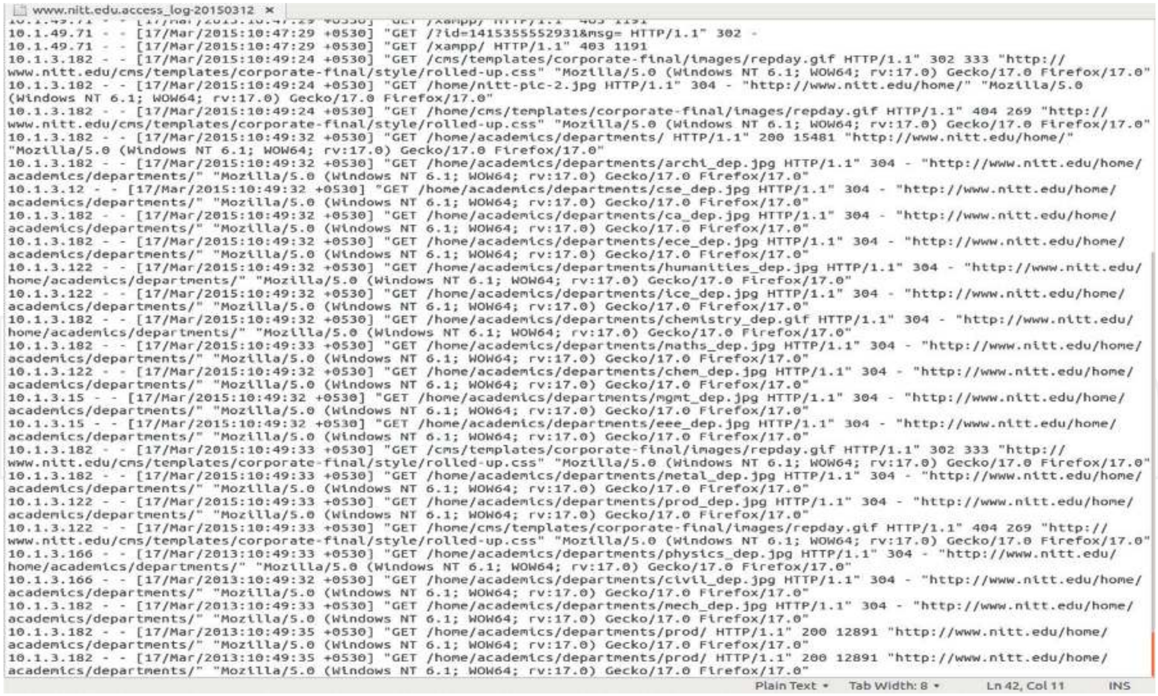
```
    03/12/2015 8:14:07 AM,"Rule ““Block Windows File Sharing”“ blocked
(192.168.1.54, netbios-ssn(139)).”,"Rule ““Block Windows File Sharing”“ blocked
(192.168.1.54, netbios-ssn(139)). Inbound TCP connection. Local address,service
is (KENT(172.30.128.27),netbios-ssn(139)). Remote address,service is
(192.168.1.54,39922). Process name is ““System”“.”
    03/12/2015 9:04:04 AM,Firewall configuration updated: 398 rules., Firewall
configuration updated: 398 rules.
```

- ***Sample Syslog Entries***

```
    Mar 1 06:25:43 server1 sshd[23170]: Accepted publickey for server2 from
172.30.128.115 port 21,011 ssh2.
    Mar 1 07:16:42 server1 sshd[9326]: Accepted password for murugiah from
10.20.30.108 port 1070 ssh2.
    Mar 1 07:16:53 server1 sshd[22938]: reverse mapping checking getaddrinfo for
ip10.165.nist.gov failed - POSSIBLE BREAKIN ATTEMPT!
    Mar 1 07:26:28 server1 sshd[22572]: Accepted public key for server2 from
172.30.128.115 port 30,606 ssh2.
    Mar 1 07:28:33 server1 su: BAD SU kkent to root on /dev/ttyp2.
    Mar 1 07:28:41 server1 su: kkent to root on /dev/ttyp2.
```

Types of log	Attacks	Tools for log analysis
DMesg log	This is not a log file, but this is used for determining anomalous activity from recent bots.	—
Debugging log	Stack tracing to determine the nature of application and service-based attacks.	—
Firewall log	Direct method for auditing the firewall.	Event Log Analyzer, event logging and monitoring services
System log	Determines if someone is trying or has executed buffer overflow.	Syslog-ng, Log & Event Manager
Network log	Determining Web-based attacks and DDoS attacks.	Splunk, Log4j2
Web server access log	Determining Web-based attacks (XSS, XSRF, SQLI), remote file inclusion, local file inclusion and flooding attacks.	Nihuo Web Log Analyzer
Web server error log	Determining Web-based attacks.	Nihuo Web Log Analyzer
VM log	Determining hypervisor-related attacks.	Virtual Machine Log Auditor, JVM controller
Authentication log	Auditing of attacks on credentials and determines the unauthorized access.	
Audit log	Determining unauthorized user access to the system and network. Includes destination addresses, user login information, and timestamp.	WP Security Audit Log, auditpol.exe
Database log	Determining database-related attacks.	Splunk, Nihuo Web Log Analyzer

Table 1.
Different types of logs, attacks, and the log analysis tool.



```
10.1.49.71 - - [17/Mar/2015:10:47:29 +0530] "GET /?id=1415355552931&msg= HTTP/1.1" 302 -
10.1.49.71 - - [17/Mar/2015:10:47:29 +0530] "GET /xampp/ HTTP/1.1" 403 1191
10.1.3.182 - - [17/Mar/2015:10:49:24 +0530] "GET /cms/templates/corporate-final/images/repday.gif HTTP/1.1" 302 333 "http://
www.nitt.edu/cms/templates/corporate-final/style/rolled-up.css" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:17.0) Gecko/17.0 Firefox/17.0"
10.1.3.182 - - [17/Mar/2015:10:49:24 +0530] "GET /home/nitt-pic-2.jpg HTTP/1.1" 304 - "http://www.nitt.edu/home/" "Mozilla/5.0
(Windows NT 6.1; WOW64; rv:17.0) Gecko/17.0 Firefox/17.0"
10.1.3.182 - - [17/Mar/2015:10:49:24 +0530] "GET /home/cms/templates/corporate-final/images/repday.gif HTTP/1.1" 404 269 "http://
www.nitt.edu/cms/templates/corporate-final/style/rolled-up.css" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:17.0) Gecko/17.0 Firefox/17.0"
10.1.3.182 - - [17/Mar/2015:10:49:32 +0530] "GET /home/academics/departments/ HTTP/1.1" 200 15481 "http://www.nitt.edu/home/"
"Mozilla/5.0 (Windows NT 6.1; WOW64; rv:17.0) Gecko/17.0 Firefox/17.0"
10.1.3.182 - - [17/Mar/2015:10:49:32 +0530] "GET /home/academics/departments/archi_dep.jpg HTTP/1.1" 304 - "http://www.nitt.edu/home/
academics/departments/" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:17.0) Gecko/17.0 Firefox/17.0"
10.1.3.12 - - [17/Mar/2015:10:49:32 +0530] "GET /home/academics/departments/cse_dep.jpg HTTP/1.1" 304 - "http://www.nitt.edu/home/
academics/departments/" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:17.0) Gecko/17.0 Firefox/17.0"
10.1.3.182 - - [17/Mar/2015:10:49:32 +0530] "GET /home/academics/departments/ca_dep.jpg HTTP/1.1" 304 - "http://www.nitt.edu/home/
academics/departments/" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:17.0) Gecko/17.0 Firefox/17.0"
10.1.3.182 - - [17/Mar/2015:10:49:32 +0530] "GET /home/academics/departments/ece_dep.jpg HTTP/1.1" 304 - "http://www.nitt.edu/home/
academics/departments/" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:17.0) Gecko/17.0 Firefox/17.0"
10.1.3.122 - - [17/Mar/2015:10:49:32 +0530] "GET /home/academics/departments/humanities_dep.jpg HTTP/1.1" 304 - "http://www.nitt.edu/
home/academics/departments/" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:17.0) Gecko/17.0 Firefox/17.0"
10.1.3.122 - - [17/Mar/2015:10:49:32 +0530] "GET /home/academics/departments/ice_dep.jpg HTTP/1.1" 304 - "http://www.nitt.edu/home/
academics/departments/" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:17.0) Gecko/17.0 Firefox/17.0"
10.1.3.182 - - [17/Mar/2015:10:49:32 +0530] "GET /home/academics/departments/chemistry_dep.gif HTTP/1.1" 304 - "http://www.nitt.edu/
home/academics/departments/" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:17.0) Gecko/17.0 Firefox/17.0"
10.1.3.182 - - [17/Mar/2015:10:49:33 +0530] "GET /home/academics/departments/maths_dep.jpg HTTP/1.1" 304 - "http://www.nitt.edu/home/
academics/departments/" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:17.0) Gecko/17.0 Firefox/17.0"
10.1.3.122 - - [17/Mar/2015:10:49:32 +0530] "GET /home/academics/departments/chem_dep.jpg HTTP/1.1" 304 - "http://www.nitt.edu/home/
academics/departments/" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:17.0) Gecko/17.0 Firefox/17.0"
10.1.3.15 - - [17/Mar/2015:10:49:32 +0530] "GET /home/academics/departments/mgmt_dep.jpg HTTP/1.1" 304 - "http://www.nitt.edu/home/
academics/departments/" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:17.0) Gecko/17.0 Firefox/17.0"
10.1.3.15 - - [17/Mar/2015:10:49:32 +0530] "GET /home/academics/departments/eee_dep.jpg HTTP/1.1" 304 - "http://www.nitt.edu/home/
academics/departments/" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:17.0) Gecko/17.0 Firefox/17.0"
10.1.3.182 - - [17/Mar/2015:10:49:33 +0530] "GET /cms/templates/corporate-final/images/repday.gif HTTP/1.1" 302 333 "http://
www.nitt.edu/cms/templates/corporate-final/style/rolled-up.css" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:17.0) Gecko/17.0 Firefox/17.0"
10.1.3.182 - - [17/Mar/2015:10:49:33 +0530] "GET /home/academics/departments/metal_dep.jpg HTTP/1.1" 304 - "http://www.nitt.edu/home/
academics/departments/" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:17.0) Gecko/17.0 Firefox/17.0"
10.1.3.122 - - [17/Mar/2015:10:49:33 +0530] "GET /home/academics/departments/prod_dep.jpg HTTP/1.1" 304 - "http://www.nitt.edu/home/
academics/departments/" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:17.0) Gecko/17.0 Firefox/17.0"
10.1.3.122 - - [17/Mar/2015:10:49:33 +0530] "GET /home/cms/templates/corporate-final/images/repday.gif HTTP/1.1" 404 269 "http://
www.nitt.edu/cms/templates/corporate-final/style/rolled-up.css" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:17.0) Gecko/17.0 Firefox/17.0"
10.1.3.166 - - [17/Mar/2013:10:49:33 +0530] "GET /home/academics/departments/physics_dep.jpg HTTP/1.1" 304 - "http://www.nitt.edu/
home/academics/departments/" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:17.0) Gecko/17.0 Firefox/17.0"
10.1.3.166 - - [17/Mar/2013:10:49:32 +0530] "GET /home/academics/departments/civil_dep.jpg HTTP/1.1" 304 - "http://www.nitt.edu/home/
academics/departments/" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:17.0) Gecko/17.0 Firefox/17.0"
10.1.3.182 - - [17/Mar/2013:10:49:33 +0530] "GET /home/academics/departments/mech_dep.jpg HTTP/1.1" 304 - "http://www.nitt.edu/home/
academics/departments/" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:17.0) Gecko/17.0 Firefox/17.0"
10.1.3.182 - - [17/Mar/2013:10:49:35 +0530] "GET /home/academics/departments/prod/ HTTP/1.1" 200 12891 "http://www.nitt.edu/home/
academics/departments/" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:17.0) Gecko/17.0 Firefox/17.0"
10.1.3.182 - - [17/Mar/2013:10:49:35 +0530] "GET /home/academics/departments/prod/ HTTP/1.1" 200 12891 "http://www.nitt.edu/home/
academics/departments/" "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:17.0) Gecko/17.0 Firefox/17.0"
```

Figure 2.
Sample access log trace as evidence.

S. No.	Fields	Value	Description
1	Remote Host	10.1.3.122	IP address of the HTTP user who makes HTTP resource request
2	Rfc931	—	Identifier used to determine client
3	Username	—	User name or user id used for authentication
4	Date: time Timezone	[17-Mar-2015: 10: 49: 33 + 530]	Date and timestamp of the HTTP request
5	HTTP request	GET/scripts/root.exe?/c+dir/HTTP/1.0	HTTP request containing (a) HTTP method—GET (b) HTTP request resource scripts/root.exe?/c+dir/ and (c) HTTP protocol version –1.0
6	Status code	200	Status of HTTP request, i.e., success or failure
7	Bytes	578	Number of bytes of data transferred during the HTTP request
8	Referral URL	https://www.nitt.edu/ OLCLD/view.php?q = book/	Referrer header of the HTTP request (containing URL of the page from which this request was initiated) if present, and "-" otherwise
9	User agent	Mozilla/4.08 [en] (Win98; I; Nav)	Browser Identification String

Table 2.
Description of the access log format.

3.2 Evidence collection from cloud storage

It is the process of collecting evidence from cloud storage such as Dropbox, Microsoft SkyDrive, Google drive, etc., using the Web browser and also by downloading files using existing software tools [11–13]. This helps to identify the illegal modification or access of cloud storage during the uploading or downloading of file contents in storage media and also checks whether the attacker alters the timestamp information in user's accounts. The Virtual Forensic Computing (VFC) tool is used by forensic investigators to identify evidence from VM image file. The evidence is accessed for each account using the Web browser running in the cloud environment by recording the encoded value of VM image. The packets are captured using network packet tools, namely Wireshark, snappy, etc., of each VM instance running in hosts. The account information is synchronized and downloaded using client accessing software of each device which is used to identify the source of evidence. The evidence is isolated from the files found in VM using "C:\Users\[username]\ Dropbox\" for Dropbox as shown in **Figure 3**. The zip file contains the name of the folder that can be accessed via the browser to determine the effect of a timestamp in a drive. If an attacker modifies the contents of a file, the evidence is found by analyzing the VM hard drive, history of files stored in the cloud, and also from a cache. It can also be analyzed by computing the hash value of the VM image. The evidence of Google Drive cloud storage is depicted in **Figure 4**.

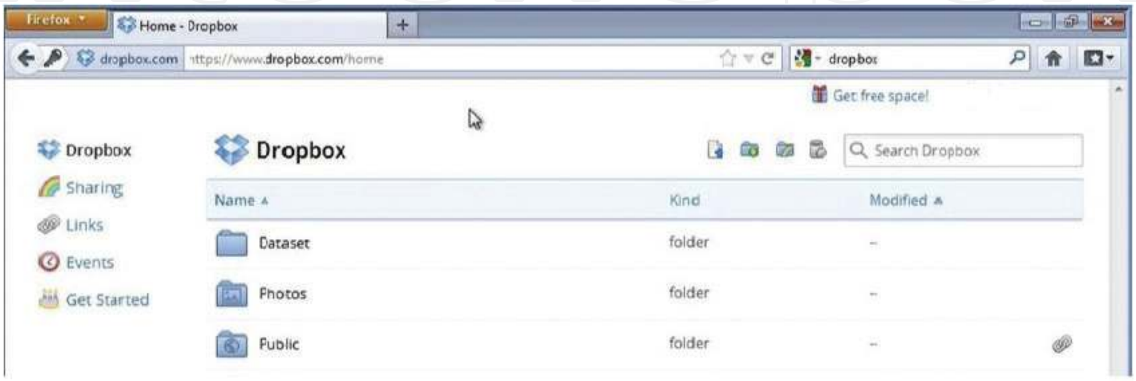
3.3 Evidence collection via a Web browser

The clients communicate with the server in the cloud environment with the help of a Web browser to do various tasks, namely checking email and news, online shopping, information retrieval, etc. [14–18]. Web browser history is a critical source of evidence. The evidence is found by analyzing the URLs in Web browser history, timeline analysis, user browsing behavior, and URL encoding, and is recovered from deleted information. Here is an example of Web browser URLs,

https://www.nitt.edu/en#files:/Documents/<Folder name>,

https://www.nitt.edu/en#files:/E:<Folder ID>.

Similarly, the evidence stored in Web browser cache at the root directory of a Web application is used to identify the source of an attack. **Table 3** indicates the evidence collection process and recovery method for various Web browsers.



Name ▲	Kind	Modified ▲
Dataset	folder	–
Photos	folder	–
Public	folder	–

Figure 3.
Dropbox evidence.

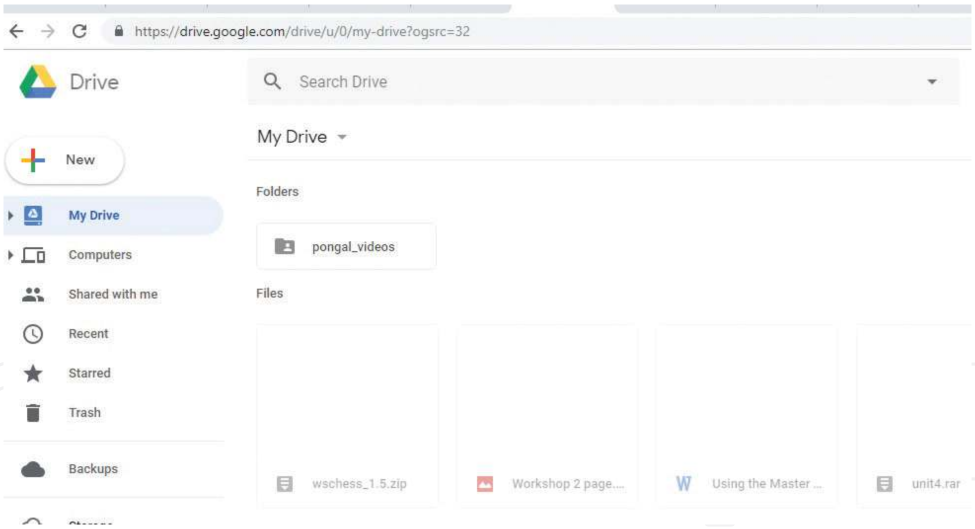
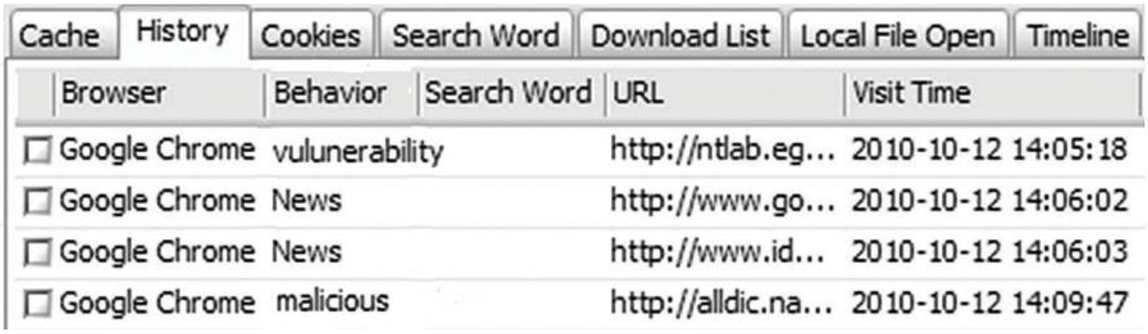


Figure 4.
Google Drive evidence.

Web browser	Information to be analyzed	Tools for forensic investigation	Recovery method for evidence identification
Internet Explorer	Index.dat History Cache Cookies	Pasco Web historian 6.13 Index.dat analyzer 2.5 Net analysis 1.52 Encase 6.3 FTK 3.3 WEFA	Recovery from internet files Analyzing the index.dat files weekly/daily history Recovery of the evidence from index.dat file through carving method Recovery from cookies
Google Chrome	Bookmark history Bookmark downloads Cookies List of search words Cache	Chrome analysis 1.0 Net analysis 1.52 Cache back 3.17 WEFA	Recovery of session file through carving method
Mozilla Firefox	History Cookies history Download list Cache Bookmarks	Firefox forensic 2.3 Net analysis 5.2 Cache back 3.17 Encase 6.3 FTK 3.3 WEFA	Recovery of cache files
Safari	History Cache Cookies	Web historian 6.13 Net analysis 1.52 Cache back 3.17 Encase 6.3 FTK 3.3 WEFA	Recovery of session files, cookies
Opera	History Cache Cookies Bookmarks	Web historian 6.13 Net analysis 1.52 Cache back 3.17 Encase 6.3 WEFA	Recovery of cookies

Table 3.
Evidence collection process and recovery method for different Web browsers.

Here is an example of a Chrome forensic tool that captures and analyzes data stored in Google Web browser. It analyzes the data from the history, web logins, bookmarks, cookies, and archived history. It identifies the evidence from C:\Users\USERNAME\Appdata\Local\Google chrome\UserData\Default. **Figure 5** depicts the Google Chrome analysis forensic tool.



Browser	Behavior	Search Word	URL	Visit Time
□ Google Chrome	vulunerability		http://ntlab.eg...	2010-10-12 14:05:18
□ Google Chrome	News		http://www.go...	2010-10-12 14:06:02
□ Google Chrome	News		http://www.id...	2010-10-12 14:06:03
□ Google Chrome	malicious		http://alldic.na...	2010-10-12 14:09:47

Figure 5.
Chrome forensic analysis tool.

Forensic analysis framework	Evidence collection for cloud storage	Evidence collection for cloud log analysis
Evidence identification	Identification of evidence from cloud storage (Dropbox, iCloud, SkyDrive and Google Drive, etc.) and also from user account information	Identification of evidence from cloud log files
Evidence collection	Collecting the evidence from VM image to access the cloud storage account, using packet analysis tools such as Ethernet cap, Wireshark tool, Burp suite, etc. to capture packets between the client and server Collecting evidence from VM browser such as Google Chrome, chromium browser, Internet Explorer, Apple Safari, Mozilla Firefox, etc. Collecting the evidence from cloud storage namely, user account and password Collecting the evidence from client software to access the VM hard drive and also to synchronize the user account to retrieve the files and folders in VMs	Collecting the evidence from various sources in VM as log files, namely network log, access log, authentication log, error log, database log, etc. and through network analysis tools such as Wireshark, Snort, Snappy tool, Burp Suite, etc.
Evidence analysis	Identifying patterns from the evidence collection process to determine the source of attacks in cloud environment	Determining the attack patterns from cloud log files and analyzing these patterns using cloud traceback mechanism to identify the source of evidence.
Evidence presentation and reporting	Forensic investigator examines the evidence and presents the evidence in court	Identifying the evidence from analysis and reporting the evidence

Table 4.
Evidence collection process for cloud forensics.

3.4 Physical memory analysis

This has the ability to provide caches of cloud computing usage that can be lost without passive monitoring such as network socket information, encryption keys, and in-memory database. They are analyzed from the physical memory dump using the “pslist” function, which recovers the process name, process identifier, parent process identifiers, and process initiation time. The processes can be differentiated using the process names ©exe© on the Windows, and ©sync© on the Ubuntu and Mac OS. **Table 4** indicates the evidence collection process for cloud forensics in cloud storage and cloud log analysis.

4. Cloud forensics challenges

This section elucidates the forensic challenges in private and public cloud. It is observed from the literature that most of the challenges are applicable to the public cloud while fewer challenges are applicable to the private cloud environment.

4.1 Accessibility of logs

Logs are generated in different layers of the cloud infrastructures [2–7]. System administrators require relevant logs to troubleshoot the system, developers need logs for fixing up the errors, and forensic investigators need relevant logs to investigate the case. With the help of an access control mechanism, the logs can be acquired from all the parties, that is, from a user, CSP, and forensic investigator.

4.2 Physical inaccessibility

The data are located in different geographical areas of the hardware device. It is difficult to access these physical access resources since the data reside in different CSPs and it is impossible to collect the evidence from the configured device. If an incident occurs, all the devices are acquired immediately in case of a private cloud environment since an organization has full control over the resources. The same methods cannot be used to access the data in case of a public cloud environment.

4.3 Volatility of data

Data stored in a VM instance in a cloud will be lost when the VM is turned off. This leads to the loss of important evidence such as syslog, network logs, registry entries, and temporary Internet files. It is important to preserve the snapshot of the VM instance to retrieve the logs from the terminated VMs. The attacker launches an attack and turns off the VM instance, hence these traces are unavailable for forensic investigation.

4.4 Identification of evidence at client side

The evidence is identified not only in the provider’s side but also the client side. The user can communicate with the other client through the Web browser. An attacker sends malicious programs with the help of a Web browser that communicates with the third parties to access the services running in the cloud. This, in turn, leads to destroying all the evidence in the cloud. One way of collecting the evidence is from the cookies, user agent, etc., and it is difficult to obtain all the information since the client side VM instance is geographically located.

4.5 Dependence of CSP trust

The consumers blindly depend on CSPs to acquire the logs for investigation. The problem arises when CSPs are not providing the valid information to the consumer that resides in their premises. CSPs sign an agreement with other CSPs to use their services, which in turn leads to loss of confidential data.

4.6 Multitenancy

In cloud infrastructures, multiple VMs share the same physical infrastructure, that is, the logs are distributed across various VMs. The investigator needs to show the logs to court by proving the malicious activities occurring from the different service providers. Moreover, it also preserves the privacy of other tenants.

4.7 Decentralization

In cloud infrastructures, the log information is located on different servers since it is geographically located. Multiple users' log information may be collocated or spread across several layers and tiers in the cloud. The application log, network log, operating system log, and database log produce valuable information for a forensic investigation. The decentralized nature of the cloud brings the challenge for cloud synchronization.

4.8 Absence of standard format of logs

Logs are available in heterogeneous formats from different layers of a cloud at CSP. The logs provide information such as by whom, when, where, and why some incidents occurred. This is an important bottleneck to provide a generic solution for all CSPs and all types of logs. **Table 5** indicates the survey of literature that deals with the challenges of cloud forensics mainly for evidence collection process.

Authors	Discussion	Forensic process
Sang et al.	Log accessibility for SaaS & PaaS	Evidence collection
Zawood et al.	Focus on the integrity of log files	Evidence collection
Dystra et al.	Log collection and accessibility of logs	Evidence collection
Thorpe et al.	VM kernel logs for forensic investigation	Log contention
Boeck et al.	Confidentiality and log integrity	Evidence collection
Zaferulla et al.	Uses Eucalyptus logs for forensic investigation	Evidence analysis
Marty et al.	Collection of logs from different cloud components	Log retention
Sibiya et al.	Uses data mining techniques to collect logs for forensic investigation	Evidence collection
Patrascu et al.	Collection of specific logs	Evidence collection
Nakahara et al.	Evidence identification from different types of logs	Evidence collection and log retention

Table 5.
Challenges of cloud forensics.

5. Forensic tools

There are many tools to identify, collect, and analyze the forensic data for investigation. Juel et al. developed the PORs tool for the identification of online archives for providing integrity and privacy of files [19]. Dykstra et al. proposed a forensic tool for acquiring the cloud-based data in management plane [6]. It ensures trust in cloud infrastructures. Moreover, Encase and Access data FTK toolkit are used for the identification of trusted data to acquire the evidence. Similarly, tools such as evidence finder and F-response are used to find the evidence related to social networks. Dystra et al. proposed FROST, an open source OpenStack cloud tool for the identification of evidence from virtual disks, API logs, firewall logs, etc. [20].

6. Open research problems in cloud forensics

Many researchers have proposed various solutions to mitigate the challenges of cloud forensics. Some of the researchers have proposed new approaches to test the attacks in real-time environment. CSPs have not adopted the proposed solutions yet. Customers or investigators rely on CSPs to collect the necessary logs since they do not have direct physical access. Customers or investigators depend on CSP to collect the various information from the registry, hard disk, memory, log files etc. Even though various forensic acquisition process is proposed still the dependence of CSP remain unsolved. The critical issue is the usage of bandwidth resources. If the cloud storage is too high, then it results in more utilization of bandwidth. There is insufficient work evolved to preserve the chain of custody to secure provenance. There is no ideal solution for cybercrime scene reconstruction and preservation of evidence. Another critical issue is based on the modification of existing forensic tools that may lose evidence. Some researchers have proposed logging as a service to provide confidentiality, integrity, and authentication [3]. This solution is not suitable for IaaS cloud.

7. Case study

This section introduces a hypothetical forensic case study related to a cloud storage service and also describes a forensic investigation of the case.

7.1 Case study: cloud storage

The organization "X" found that their document named as "X_new.pdf" about the new release of a product has been leaked to their competitor [21–24]. "Mr. Morgan" was managing the credential files of the document stored in the cloud. At the initial stage of the investigation process, the suspect of the leaked file case was "Mr. Morgan." The forensic investigator has to identify the suspect by checking the organization network, or by the analysis of log files, or by collecting the trace of relevant file in the network. Mr. Morgan's network does not have any clue about the secrets since he uses only the personal computer (PC) and Android phone for business. To identify the suspects, the forensic investigator seized the PC and Android phone since these are the target devices used by the adversary. From the suspected devices, the leaked file has not been detected. Later, the investigator started analyzing the unallocated area in the file system, operating system, external devices such as hard drive, tablets, etc., and the Web service, but no evidence was found in the investigation. The investigator found that the Dropbox was installed in the PC and five files of config.db have been accessed recently. The forensic investigator issued the search warrant and identified the evidence in Dropbox by accessing Morgan's Dropbox storage with the username and password. It was observed that Morgan recently uploaded five files in Dropbox and identified that one of the files named as "XYZ_new.pdf" had the same contents as "X_new.pdf." Later, he deleted the traces of uploading or downloading the contents in PCs. The investigator found that Mr. Morgan has deleted the traces of the file contents and shared the evidence stored as "XYZ_new.pdf" with the competitor through an external SD card.

7.2 Case study: online railway ticket fraud

An online railway ticket booking service provider claimed that some unknown user had used the internet ticket booking facility to book 44 railway tickets using the stolen credit card details [25]. It has been charged back from the credit card companies for all transactions which led to a huge loss to the service provider. It is inferred from the investigation that the suspects have booked 44 tickets with different names of a person through the website at different locations. Later on, through investigation, the investigator found that the suspects arrived from a particular IP address, thereby seized the contents of the user accounts with the password, and the stolen credit cards were recovered from the suspects.

7.3 Case study: morphed photographs

The user got threatening pornographic emails from the adversary that one photograph was posted on the popular website [25]. The IP address for posting such threatening emails on the website was retrieved and was traced to a company. During an investigation, it was observed that the emails were sent from the company premises from one of the terminals. The log records and cookies were examined from the seized system and the morphed photographs were found in one of the systems used by the suspect. The mirror image of the hard disk was collected and analyzed using disk imaging and forensic analysis tools to recover all the data files required for the case. At the end of the investigation, it was found that the suspect was an ex-colleague of the company.

7.4 Case study: malicious insiders

Mr. X is an intruder who intends to exploit victims by sending malicious Web page in the cloud [26]. He uses a vulnerability to exploit the cloud presence of Buzz Coffee, a legitimate company. He installs a rootkit that injects a malicious payload into Web pages displayed and hides the malicious activity from the operating system. It redirects victims to the website, which infects them with malware. The users complain to the legitimate company that they are being infected, so the company went to the investigator to investigate the case by finding all the traces of the malicious Web page to identify the malicious user.

7.5 Case study: ransomware attack

A securities and brokerage firm became a victim of a ransomware attack [26]. The hacker demanded a ransom of two Bitcoins for each system that was infected. During the investigation process, it was observed that several other critical systems were infected with the same ransomware. Emails with malicious attachments appeared to be originating from a foreign location and were identified as the source of infection. The organization decided to take a proactive approach toward security with the focus on real-time monitoring to thwart such attacks in the future.

8. Conclusion

Cloud computing offers on-demand services (CPU, memory, network bandwidth, storage, applications, etc.) to users by allocating virtual instances and software services. Security is a major concern in the cloud wherein investigation of security attacks and crimes are very difficult. Due to the distributed nature of attacks and crimes in cloud, there is a need for efficient security mechanism. As cloud logs are spread across different virtual/physical machines (VM instances), switches, routers, etc., and also the customer (end user) is not aware of the activities of VM instances, cybercriminals exploit these sources to exhaust all the resources running in the cloud. Hence, evidence collection plays a crucial role to identify the suspects. However, collecting logs from the cloud infrastructure is extremely difficult because the investigator/security analyst has to depend on CSPs for collecting the logs and they have little control over the infrastructure. So, in order to identify the suspicious activity involved in the cloud, this chapter surveys the various forensic processes, evidence collection techniques for cloud forensics and the various challenges faced in cloud environment for forensic investigation.

Conflict of interest

The author does not have any conflict of interest.

Author details

Thankaraja Raja Sree* and Somasundaram Mary Saira Bhanu
Department of Computer Science and Engineering, National Institute of Technology, Tiruchirappalli, India

*Address all correspondence to: trajasree87@gmail.com

References

[1] Mell P, Grance T. The NIST Definition of Cloud Computing. Computer Security Division, Information Technology Laboratory, National Institute of Standards and Technology, United States Department of Commerce. Gaithersburg. 2011:1-7 DOI: Special Publication 800-145

[2] Liu F, Tong J, Mao J, Bohn R, Messina J, Badger L, et al. NIST Cloud Computing Reference Architecture. Gaithersburg: NIST Special Publication. 2011. pp. 1-28. DOI: NIST SP 500-292

[3] Pichan A, Lazarescu M, Soh ST. Cloud forensics: Technical challenges, solutions and comparative analysis. Digital Investigation. 2015;**13**:38-57. DOI: 10.1016/j.diin.2015.03.002

[4] Guo H, Jin B, Shang T. Forensic investigations in cloud environments. In: 2012 International Conference on Computer Science and Information Processing (CSIP); 2012 Aug 24; IEEE. 2012. pp. 248-251. DOI: 978-1-4673-1411-4/12/

[5] Zawoad S, Hasan R. Cloud forensics: A meta-study of challenges, approaches, and open problems. 2013. arXiv preprint: 1302.6312

[6] Dykstra J, Sherman AT. Acquiring forensic evidence from infrastructure-as-a-service cloud computing: Exploring and evaluating tools, trust, and techniques. Digital Investigation. 2012;**9**:S90-S98. DOI: 10.1016/j.diin.2012.05.001

[7] Marty R. Cloud application logging for forensics. In: Proceedings of the 2011 ACM Symposium on Applied Computing; 2011 Mar 21; ACM. 2011. pp. 178-184. DOI: 10.1145/1982185.1982226

[8] Anwar F, Anwar Z. Digital forensics for eucalyptus. In: 2011 Frontiers of Information Technology; 2011 Dec 19; IEEE. pp. 110-116. DOI: 10.1109/FIT.2011.28

[9] Khan S, Gani A, Wahab AW, Bagiwa MA, Shiraz M, Khan SU, et al. Cloud log forensics: Foundations, state of the art, and future directions. ACM Computing Surveys (CSUR). 2016;**49**(1):7. DOI: 10.1145/2906149

[10] Kent K, Souppaya M. Guide to Computer Security Log Management. Gaithersburg: NIST Special Publication; 2006. p. 92. DOI: N060928K

[11] Zhang OQ, Kirchberg M, Ko RK, Lee BS. How to track your data: The case for cloud computing provenance. In: 2011 Third IEEE International Conference on Cloud Computing Technology and Science; 2011 Nov 29; IEEE. -453. DOI: 446, 10.1109/CloudCom.2011.66

[12] Quick D, Choo KK. Forensic collection of cloud storage data: Does the act of collection result in changes to the data or its metadata? Digital Investigation. 2013;**10**(3):266-277. DOI: 10.1016/j.diin.2013.07.001

[13] Chung H, Park J, Lee S, Kang C. Digital forensic investigation of cloud storage services. Digital Investigation. 2012;**9**(2):81-95. DOI: 10.1016/j.diin.2012.05.015

[14] Nepal S, Ranjan R, Choo KK. Trustworthy processing of healthcare big data in hybrid clouds. IEEE Cloud Computing. 2015;**2**(2):78-84. DOI: 10.1109/MCC.2015.36

[15] Yusoff MN, Dehghantanha A, Mahmod R. Forensic investigation of social media and instant messaging services in Firefox OS: Facebook, Twitter, Google+, Telegram, OpenWapp, and Line as case studies. In: Contemporary Digital Forensic Investigations of Cloud and Mobile

Applications. 2017. pp. 41-62. DOI: 10.1016/B978-0-12-805303-4.00004-6

[16] Norouzizadeh Dezfouli F, Dehghantanha A, Eterovic-Soric B, Choo KK. Investigating social networking applications on smartphones detecting Facebook, Twitter, LinkedIn and Google+ artefacts on Android and iOS platforms. Australian Journal of Forensic Sciences. 2016;**48**(4):469-488

[17] Quick D, Choo KK. Google drive: forensic analysis of data remnants. Journal of Network and Computer Applications. 2014;**40**:179-193. DOI: 10.1016/j.jnca.2013.09.016

[18] Oh J, Lee S, Lee S. Advanced evidence collection and analysis of web browser activity. Digital Investigation. 2011;**8**:S62-S70. DOI: 10.1016/j.diin.2011.05.008

[19] Juels A, Kaliski BS Jr. PORs: Proofs of retrievability for large files. In: Proceedings of the 14th ACM conference on Computer and Communications Security; ACM. 2007. pp. 584-597. DOI: 10.1145/1315245.1315317

[20] Dykstra J, Sherman AT. Design and implementation of FROST: Digital forensic tools for the OpenStack cloud computing platform. Digital Investigation. 2013;**10**:S87-S95

[21] Chung H, Park J, Lee S, Kang C. Digital forensic investigation of cloud storage services. Digital investigation. Nov 1 2012;**9**(2):81-95. DOI: 1709/1709.10395

[22] Daryabar F, Dehghantanha A, Choo KK. Cloud storage forensics: MEGA as a case study. Australian Journal of Forensic Sciences. 2017;**49**(3):344-357. DOI: 10.1080/00450618.2016.1153714

[23] Martini B, Choo KK. Cloud storage forensics: ownCloud as a case study. Digital Investigation. 2013;**10**(4): 287-299. DOI: 10.1016/j.diin.2013.08.005

[24] Teing YY, Dehghantanha A, Choo KK, Yang LT. Forensic investigation of P2P cloud storage services and backbone for IoT networks: BitTorrent Sync as a case study. Computers & Electrical Engineering. 2017;**58**:350-363. DOI: 10.1016/j.compeleceng.2016.08.020

[25] http://prateek-paranjpe.blogspot.com/p/cyber-forensics-case-studies.html

[26] https://webforms.ey.com/Publication/vwLUAssets/ey-responding-to-cybercrimeincidents-in-india-new/$FILE/ey-responding-to-cybercrime-incidents-in-india.pdf