# Practical Attacks against Virtual Desktop Infrastructure (VDI) Solutions

**Dan Koretsky, Sr. Security Strategist**

**Lacoon Mobile Security**

## Introduction

Enterprises are increasingly adopting Bring Your Own Device (BYOD) initiatives. In order to address the security and privacy concerns of mobility in the enterprise, security professionals together with IT, legal and even management teams measure how various processes, methodologies, and technologies weigh one against the other.

The Virtual Desktop Infrastructure (VDI) technology is considered one such practical solution. Since VDI provides a remote workstation offering so that no data is stored locally on an endpoint device, it is touted as the security solution against data theft. While such a solution comes in handy when a mobile device is stolen, how does VDI scale when the device itself is compromised by a threat actor? With mobile devices acting as a penetration vehicle into the enterprise and its resources, and threat actors progressively threatening this platform, such a question must be raised. In fact, as this paper shows, device compromise is a real and practical threat that enterprises must take into consideration.

In this paper, we examine the architecture of VDI and analyze its benefits and shortcoming as a security solution. Looking at both iOS-based and Android-based devices, we consider various attack vectors threat actors use to bypass the VDI solutions and efficiently glean sensitive and confidential corporate information.

It is important to note that this article does not look at one VDI implementation as opposed to the next in terms of security. We do not test for vulnerabilities in implementation or provide vulnerability exploits that threat actors can later use. Rather, all the attack vectors that we present leverage potential problems with any VDI solution such as extracting passwords from the application's memory or scraping the screen contents.

The aim of this article is to provide enterprises with a comprehensive secure mobile adoption strategy that can be easily applied. As such, we also take a brief look at other current BYOD solutions and show how using these existing technologies, enterprises can integrate mobile security within their overall security strategy.

# Threats to Mobile VDI Implementations

## Threat #1: Using an mRAT for its Keylogging Capabilities

Mobile Remote Access Trojans (mRATs) are mobile surveillance software installed on particular individuals' devices. As their name implies, mRATs are privy to all data on the mobile and all communications passed on the device, as well as capable of manipulating mobile resources.

As opposed to the mass malware apps, such as premium SMS-grabbing malware or common banking Trojans distributed en masse with the hopes of any unsuspecting user falling for a scam, mRATs are much more target-focused and more persistent. Accordingly, threat actors invest heavily in discovering, creating and developing new techniques to install and hide mRATs on the user's device.

mRATs used to target the organization, typically do this for cyber-espionage purposes. Consequently, the impact of such a threat on the organization is extremely high – from gaining access to corporate emails and exfiltrating memos discussing the company's roadmap, to recordings of confidential phone calls and board meetings.

It is important to note that mRATs are not used only against high-end targets. Private individuals have too been known to be victims of mRATs, for example, in the case of cheating spouses. However, a compromised device within the organization, regardless of the threat actors' motivations, still suffers from the same impact – whether data leakage or breach or regulation. In the former case, consider the consequence of an mRAT installed on a military official or on a salesperson device accessing the victim's contact list. In the latter case, an mRAT accessing a health provider's main servers may eventually lead to a HIPAA-regulatory breach.

### mRATs Capabilities

mRATs that break VDIs typically consist of the following capabilities which may prove to be costly to the business:

- **Keylogging.** Examples: Any keyboard activity, from the typing of passwords to the VDI server to the authoring of M&A-related emails, is recorded by an external party.
- **Screen Scraping.** Examples: Any activity that appears on the screen such as customer data is photographed by an external party.
- **Collecting passwords.** Examples: corporate email credentials and corporate-customized applications, as well as CRM, ERP and other Cloud-based services.

### The Range of mRATs

Lacoon's Mobile Research Team identified more than 50 families of mRATs. These mRATs run the gamut from dedicated high-end groups targeting specific organizations and activists, to low-end software targeting the private consumers.
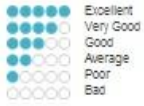
Publicized recent examples of mRATs from the high-end of the spectrum include:

- DaVinci Remote Control System (RCS), by the Hacking Team (June 2014) – Recent revelations showed that this software was installed in about a dozen countries, targeting more than 30 activists and journalists. Promoted as surveillance software for Android and iOS-based devices, RCS intercepts SMS/ MMS messages, takes camera snapshots and records all video.

- Careto – "The Mask" (2008 - January 2014) – A malware campaign found in 31 countries targeting government institutions, and energy, oil and gas companies via a cross-platform malware toolkit. Careto leveraged high-end exploits, a sophisticated mRAT, a rootkit, a bootkit, Mac OS X and Linux versions as well as potential versions for Android iOS.

- KorBanker (November 2013) – Mobile malware which targeted 6 South Korean banks. In this campaign, a fake app impersonated the Google Play store app that, once installed, further installed a second malicious mobile app. The malicious app replaced any previously installed official banking app with a fake banking app capable of stealing user credentials.

- FinSpy, by The Gamma Group (August 2012, March 2013) – Reportedly used by law enforcement agencies targeting journalists and civilian activist groups worldwide. FinSpy can turn on the mobile's microphone, take screenshots and bypass encryption methods and communications. FinSpy infected mobile devices using spear-phishing emails, and according to forensics results, utilized exploitation capabilities for both iOS and Android.

- WUC's Conference (March 2013) –Android-based mRAT which targeted Tibetan activists. Threat actors sent conference attendees a spear-phishing email containing the mRAT. The mRAT was capable of collecting contacts, call logs, geo-location data and SMS messages.

- SD-Card malware (February 2013) – Users downloading Google apps which masqueraded as clean up tools were hit with audio-recording malware upon mobile sync with the their PC.

- SpyEra (April 2012) – This malware is seemingly one more mass-distributed malware masking as a game featured in the app market. However, a closer look shows that its capabilities are that of an mRAT and is very dedicated. It can be assumed that the threat actors plan to mass-distribute the malware as to eventually hit the right target.

At the lower end of the spectrum are mRATs which most commonly portray themselves as promoting parental controls and spouse monitoring. The operators of these mRATs follow a SaaS business model where the exfiltrated data is stored and managed as a dedicated Cloud service. Similarly to a well-run business, the operators of these tools promise professional world-wide support. Their GUI is simple and user-friendly to enable all users – from the tech-savvy to the technologically impaired – to run their service.

The difference between the military and non-military grade mRATs? The device infection vectors and accordingly, their cost. Current estimates hold mRATs in the former category at

$350K[1]. Meanwhile, the lower-end of mRATs follow a monthly low licensing model– sometimes as low as $4.99.The amazing part is that the end-result is essentially the same on the targeted devices. So for just a bit more than the price of a Starbucks latte, a threat actor can purchase a mRAT with nearly identical capabilities to that of a top-end mRAT.

## CELL PHONE SPY SOFTWARE REVIEWS

| RANK | 1 MSPY | 2 MOBI STEALTH | 3 SPYBUBBLE | 4 STEALTH GENIE | 5 EBLASTER MOBILE | 6 FLEXISPY | 7 MOBILESPY | 8 HIGHSTER MOBILE | 9 SPYERA | 10 SPY PHONETAP |
|---|---|---|---|---|---|---|---|---|---|---|
| Visit WebSite | GO | GO | GO | GO | GO | GO | GO | GO | GO | GO |
| Review | READ REVIEW | READ REVIEW | READ REVIEW | READ REVIEW | READ REVIEW | READ REVIEW | READ REVIEW | READ REVIEW | READ REVIEW | READ REVIEW |

Rating legend: ●●●●● Excellent, ●●●●○ Very Good, ●●●○○ Good, ●●○○○ Average, ●○○○○ Poor, ○○○○○ Bad

### FEATURES

| Feature | 1 MSPY | 2 MOBI STEALTH | 3 SPYBUBBLE | 4 STEALTH GENIE | 5 EBLASTER MOBILE | 6 FLEXISPY | 7 MOBILESPY | 8 HIGHSTER MOBILE | 9 SPYERA | 10 SPY PHONETAP |
|---|---|---|---|---|---|---|---|---|---|---|
| OS Support | Android, iPhone, iPad, BlackBerry, Symbian Nokia, Windows Mobile | Android, iPhone, BlackBerry, Symbian, Windows Mobile | iPhone, Android, BlackBerry, Windows Mobile, Symbian | Android, iPhone, BlackBerry | Android, BlackBerry | Android, iPhone, BlackBerry, Symbian, Windows Mobile | iPhone, Android, BlackBerry, Windows Mobile, Symbian | iPhone, BlackBerry, Android, Symbian S60, Nokia, Windows Mobile | iPhone, iPad, BlackBerry, Android, Symbian, Windows Mobile | iPhone, Android, BlackBerry, Nokia phone, Windows Mobile |
| SPY on Calls | ✓ | ✗ | ✓ | ✓ | ✗ | ✓ | ✓ | ✓ | ✓ | ✓ |
| SPY on SMS and MMS | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| SPY on Emails | ✓ | ✗ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✗ |
| Track GPS Location | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Monitor Internet Use | ✓ Browsing History, Website Bookmarks, Blocking Websites | ✓ Browsing Website History | ✓ URL Tracking | ✓ Browsing History | ✓ Browsing History | ✗ | ✓ Browsing History | ✗ | ✗ | ✗ |
| Access Address Book | ✓ | ✗ | ✓ | ✓ | ✗ | ✓ | ✓ | ✓ | ✓ | ✗ |
| Access Calendar | ✓ | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |
| Instant Messages | ✓ Skype, WhatsApp, iMessage | ✓ Skype | ✗ | ✓ Skype, Gtalk, BBM | ✓ BlackBerry Messenger chats | ✓ WhatsApp, BlackBerry Messages | ✓ | ✗ | ✓ BBM, Facebook chat | ✗ |

*A screenshot found on a website reviewing the Top 10 "Cell Phone Monitoring Software Review", March 2014*
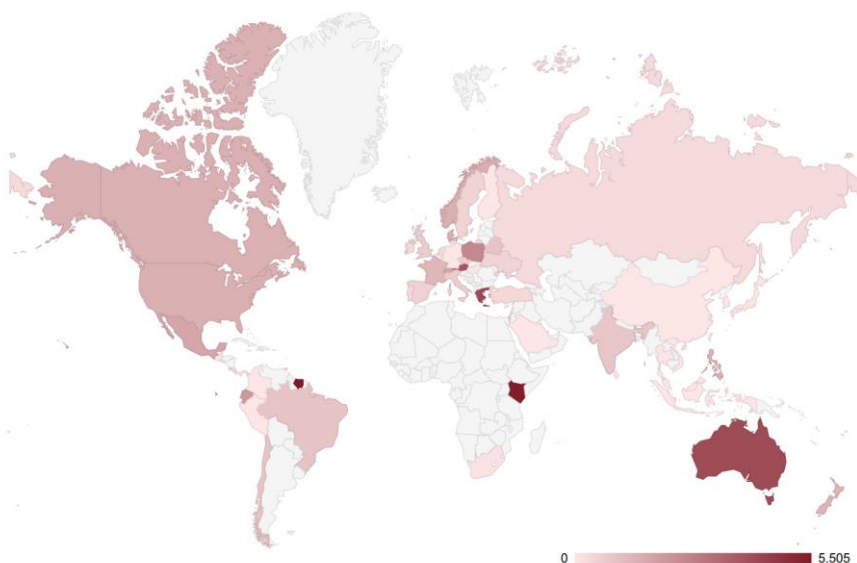
## mRATs in the Enterprise

To paint a better picture of how common mRATs are in the enterprise, Lacoon Mobile Security partnered with CheckPoint Software Technologies, the leading network-firewall vendor. The joint research enabled us to sample mobile devices communicating through corporate WiFi access points that are connected to the CheckPoint firewall. Based on data extracted from 95 gateways, which roughly represent 90 enterprises, we were able to focus on infected devices within the enterprise, analyze the communications to the C&C servers and the data that the threat actors gathered from users' mobile devices.
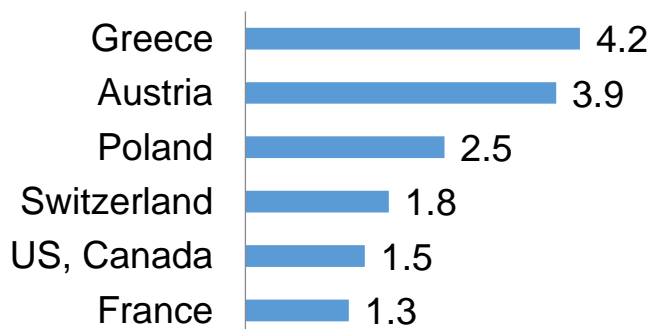
---

[1] http://bits.blogs.nytimes.com/2012/08/13/elusive-finspy-spyware-pops-up-in-10-countries/

## Latest Survey Findings

- **Number of Infected Devices:** 7700 (estimated infection rate of 0.12%)
- **mRATs Used:** at least 16 different mRATs were identified. Most commonly used: MSpy, Spy2Mobile, My Mobile Watchdog, Bosspy, MobiStealth and TalkLog
- **For gateways with 2000 devices or more, 50% have infections.**
- **Type of Infected Operating Systems:** 42% are iOS-based and 58% android based
  - **Country Infection Rates:** mRATs were found in enterprises across 48 countries. Countries included: Israel, US, Mexico, Brazil, The Netherlands, Italy, India, France and Australia. The following diagram presents the distribution of infections:



**Infections (for 1000 devices)**



| Country | Infections |
|---|---|
| Greece | 4.2 |
| Austria | 3.9 |
| Poland | 2.5 |
| Switzerland | 1.8 |
| US, Canada | 1.5 |
| France | 1.3 |

## Threat #2: Man-in-the-Middle (MitM)

The threat of Man-in-the-Middle (MitM) has always been a concern for mobile devices that are not on trusted networks. Additionally, typical alert and warning signs that individuals are used to noticing on PCs and laptops are much more subtle in their mobile counterparts.

Standard ways to defeat the threat of MitM include adopting SSL as an end-to-end encryption on most communication to remote devices. While these solutions should protect credentials and data from common threats such as malicious WiFi Access Points, they don't help against more complicated threats, such as SSL spoofing. SSL spoofing requires installing a malicious Certificate Authority (CA) on a device and having the the threat actors re-route all traffic through their servers. Although SSL pinning - hardcoding the certificate used to sign on traffic from a specific data source - can be used to mitigate such an attack, there are many situations where it isn't practical.

In an iOS-based world, a **configuration profile** is a practical attack vector that can be used to carry out MitM. A configuration profile is an extremely sensitive optional configuration file which allows to re-define different system functionality parameters such as mobile carrier settings, Mobile Device Management (MDM) settings and network settings. A user may be tricked to download a malicious configuration profile and by doing so, unknowingly provides the rogue configuration the ability to re-route all traffic from the mobile device to a server controlled by the threat actor, install a malicious CA, and even install rogue apps. This enables the threat actor to decrypt the traffic that is routed through their server, using the malicious CA, and steal credentials or other sensitive information from applications that do not provide certificate pinning.

For Android the threat is similar if the threat actor is able to first lure the device owners to install a malicious CA on their phone and then setup a method for the network interception (such as a VPN or Proxy setup on the device).

## Myth-Busting the Security of VDI Solutions

Virtualized Desktop Infrastructure (VDI) solutions provide a remote workstation offering so that no data is stored locally on an endpoint device. By minimizing the data distribution VDI solutions are commonly used by enterprises as part of their mobile adoption strategy to protect against the threat of data leakage and theft. For example, VDI solutions are popular amongst healthcare providers required to ensure patient privacy rights as well as comply with the HIPAA regulation.

The problem? There is the presumption that VDI solutions also secure the device against data exfiltration performed by threats such as mRATs and MitM as described in the section above.

## A Short Primer to VDI

There are two major mobile VDI server players – Citrix and VMware. Citrix offers also a VDI client. VMware, on the other hand, has an API so that vendors can develop their own VDI client that integrates with the VMware protocol and platform. Some client-side VDI vendors include Pivot3, Quest Software, Oracle, RedHat, MokaFive, Virtual Bridges, NComputing, Deskton and Unidesk.

**How does a common VDI architecture look like?**

The VDI server holds multiple virtualized instances to which the VDI client requests access. On the other hand, the VDI client app requests connection to an instance. If the request is granted, then the server opens a VDI connection with the client and all succeeding information is passed within that session.

It is important to point out that all communication between the VDI client app and server occurs within an encrypted SSL tunnel.

The following steps depict the flow:

1. The device owner manually inserts the VDI credentials in the VDI client app
2. The VDI client apps sends the user's credentials to the server
3. If the client is authorized, then the server authenticates the user and accordingly, sends the client the relevant list of services (also called instances or files), as well as the organizational policy for that user
4. The client app requests the server the particular service to use
5. The server opens the VDI connection. The VDI connection includes information such as the monitor, keyboard, mouse, etc.



*Overview of VDI server-client communication*

## Practical Threats against VDI

In this paper we present a spectrum of threats against VDI solutions, covering both iOS and Android-based devices.

It is important to note that although we demonstrate different threats against particular platforms, these threats do not exploit a VDI vulnerability or compromise a particular VDI solution. Rather, all the attack vectors that we present leverage potential problems with any

VDI solution, such as extracting passwords from the application's memory, and demonstrate how these assumptions can be manipulated by threat actors.

**Threat Demo #1: Using a Widely Popular mRAT on an Android-based Device**

Let's investigate one of the more common surveillance softwares against Android-based devices, mSpy.

According to our Lacoon-Checkpoint "mRATs in the Enterprise" survey, mSpy was the most popular commercial surveillance software and was detected in 19 different countries, including but not limited to, The United States, Britian, and France.

In fact, mSpy is an off-the-shelf mobile monitoring tool which can be purchased online for less than $50. mSpy provides a range of built-in capabilities and does not require any technical knowledge in order to operate. Taking a close look at its capabilities, we can see that it also offers a keylogging feature. Meaning, any data that a user types into the device, is captured by mSpy and sent to the Command and Control (C&C) server.
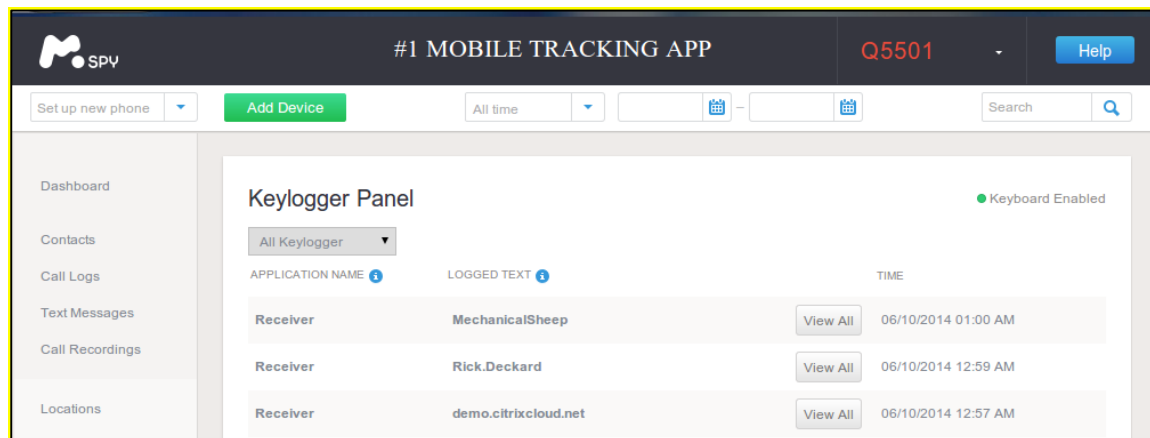


*User's VDI password as displayed through mSpy's keylogging feature*

Using the keylogging feature, it is simple to bypass any VDI solution. The mRAT does this by grabbing the employee's credentials as the employee types them into the VDI system and sends the credentials back to the threat actor. At any later stage, the threat actor can then impersonate the employee when authenticating to the VDI solution.

But the mRAT does not necessarily need to stop at stealing the credentials. In fact, the mRAT can grab any information typed into the VDI during the session - from emails to documents and presentations.

**Threat Demo #2: Man-in-the-Middle against iOS**

To demonstrate this threat, we install a malicious configuration profile on iOS. This configuration profile sets up a malicious Certificate Authority (CA) on the device and routes all the communication on the device through our demo servers. Accordingly, we show how it is possible to perform SSL spoofing on the targeted VDI client.

For demonstration purposes, we look at the Citrix Receiver VDI server and client:

www.lacoon.com

1. The victim is presented with a link to a malicious configuration profile. This link can be pushed to the users in any way that involves social engineering
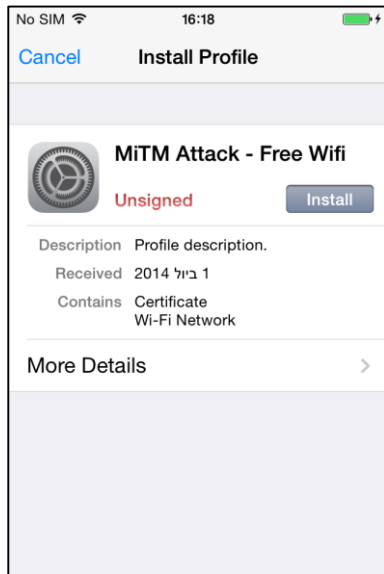


*Malicious profile containing CA*

2. At a certain stage, the victim runs the VDI client. Behind the scenes, the VDI client logs into the VDI server

3. Since we are able to see the data in the SSL connection we can see the credentials used to login to the server
   - In Citrix Receiver, the password is not sent in cleartext, but encoded using a variant of a running xor



*Extracted password from a simple modified MitM server*

We have presented one possible, and simple use, for an MitM attack against the VDI client. More complicated scenarios involve extracting the actual images and keypresses being passed between the server and client - we leave these as an exercise to the reader.

As mentioned earlier, all communications between the VDI client app and the server are tunneled through an SSL connection, and thus seemingly secure from MitM. It is important to note that the information sent in these communications are not encrypted with an additional layer as they rely on the strength of SSL. While in theory this would provide the necessary encryption - it doesn't protect against advanced threats which can bypass SSL with a malicious CA, as shown above. A solution to mitigate against the decryption of SSL would be to use SSL pinning. However, this solution is not practical in enterprise scenarios.

## Threat Demo #3: Developing a mRAT that Grabs Locally Stored Credentials on Android

Although this threat is dependent on the VDI-client implementation, we use Citrix Receiver VDI client and server for demonstration purposes:

1. The threat actor runs a privilege escalation exploit without any identifiable root marks that can be identified by the device's owner. There are numerous in the wild privilege escalation exploits that were released over the past year, readily available for any threat actor to grab. These include, but are not limited to, TowelRoot (CVE-2014-315) and VRoot (CVE-2013-6282)
2. Once the mRAT gains root privileges, the threat actor enables enable *jdwp* debugging on all the apps installed on the device
3. The mRAT connects as a debugger to the VDI client
4. The mRAT sets a breakpoint on a function that handles the credentials and sends them back to the C&C server
   - For Citrix Receiver, we used the function com.citrix.client.pnagent.asynctasks.DownloadIcaFileAndLaunchEngineTask.get IcaFileStream
     This function constructs an authentication packet. Accordingly, the VDI login credentials are passed within the function as parameters.

```
Initializing jdb ...
> stop in com.citrix.client.pnagent.asynctasks.DownloadIcaFileAndLaunchEngineTask.getIcaFileStream
Set breakpoint com.citrix.client.pnagent.asynctasks.DownloadIcaFileAndLaunchEngineTask.getIcaFileStream
>
Breakpoint hit: "thread=<15> AsyncTask #2",
com.citrix.client.pnagent.asynctasks.DownloadIcaFileAndLaunchEngineTask.getIcaFileStream(), line=138 bci=0
<15> AsyncTask #2[1] locals
Method arguments:
inName = "citrixcloud:WwCo Company Overview"
launchUrl = instance of java.net.URL(id=830045825864)
Local variables:
...
userName = "Rick.Deckard"
password = instance of char[4] (id=830041554744)
domain = "citrixcloud"
taskResult = instance of com.citrix.client.pnagent.asynctasks.results.AsyncTaskResult(id=830046472704)
<15> AsyncTask #2[1]
<15> AsyncTask #2[1] dump password
 password = {
d, e, m, o
}
```

*Debugging the session against the VDI client*

## Threat #4: Screen Scraping against Android

While the above threats effectively steal the employee's VDI credentials to enable the impersonation of a legitimate user, these attacks don't target the actual data presented from the VDI client.
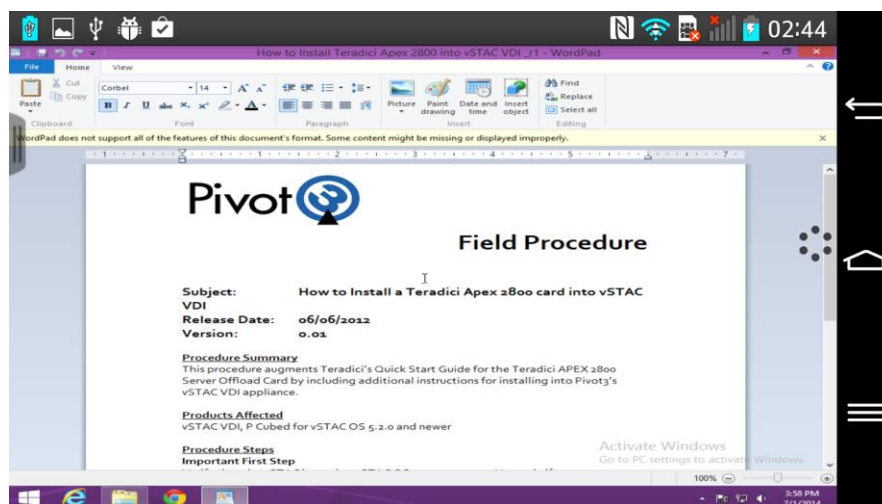
The more generic approach, screen scraping, does not require knowledge of the specific underlying VDI solution. Yet, it enables the threat actor to steal data.

There are two methods to carry out screen scraping.

1. Leverage the clipboard access support, implemented in most VDI solutions.
2. Record the screen automatically when the mRAT detects that the VDI client is connected.

We demonstrate this threat using Pivot 3's VDI solution that uses VMware's Horizon View as the VDI client:

1. The attacker runs a privilege exploit as specified in an earlier threat description
2. The mRAT monitors the current foreground activity using standard Android APIs. Data extraction is triggered only for relevant apps, such as VDI clients
3. The mRAT starts recording the screen and extracts the screen recording from the device
4. The mRAT sends the VDI client a set of key strokes, based on certain rules, to cause the content to be copied from the file to the clipboard.
   - For this example, we immediately send Ctrl+A, Ctrl+C keys to the VDI client when a file is opened. This causes a very brief visual artifact but will copy the contents of the file into the clipboard
5. The mRAT extracts the content from the clipboard and sends it the C&C


*Inside the VDI Client*

www.lacoon.com

*Data extracted from the VDI client*

## Building the Necessary Mobile Security Strategy

As demonstrated, VDI solutions cannot secure the device against data exfiltration performed by threats such as mRATs and MitM.

So, how can enterprises defend against such attacks?

1.  Look at <u>all the different threat vectors</u> that threat actors can use to exploit mobile devices to ensure nothing goes undetected. Correlate and analyze all the information from:

    a.  **Devices.** Continuous monitoring of the operating system (Android, iOS), including processes, configurations and vulnerable libraries that could impact the security stance of a device.

    b.  **Applications.** Understanding the behaviors and intent of applications (including the interfaces) on specific devices to identify immediate and long-term risky activities (e.g. time bombs); applications downloaded from "official" markets (e.g. Google Play, iTunes Store), as well as those that have been repackaged and side–loaded. Behavioral App Reputation technologies are best positioned to address this gap, but one must make sure that they are capable of detecting unknown keyloggers, screen scrapers and packaged privilege escalation exploits

    c.  **Network Connections.** Identifying rogue access points or compromised connections, as well as recognizing anomalous network traffic to and from a device that indicates an exploit.

2. <u>Accurately classify</u> low level <u>threats</u> (that have no implication on corporate assets) and more targeted advanced threats to enable appropriate responses and effective risk mitigation.

3. <u>Provide proactive threat remediation as part of a Risk Based Mobile Management (RBMM) approach -</u> be able to mitigate the threat on the device, in the network, and most importantly by risk score loopback to the VDI or container service, to block access to corporate resource when (and only when) the device is compromised.

To mitigate MitM-related threats, we call out to VDI vendors to introduce a robust framework for certificate validation and pinning to avoid unauthorized interception of the authentication and communication protocols.

## Conclusions

VDIs are certainly a beneficial tool to minimize the storage of data on a local device and consequently, the exposure of confidential data due to device theft. However, threats against the underlying VDI platform are fairly easy to carry out by using widely-distributed free tools.

The point is to recognize that VDI depends on the integrity of the host system. This means that as long as the device is uncompromised the solution protects the data. On the other hand, once the underlying platform is compromised, so does the VDI solution. In order to undermine the security of VDI solution, it is enough for the threat actor to target the device itself.

Unfortunately, as demonstrated throughout this whitepaper, compromisable devices are increasingly being introduced into the enterprise. Enterprises need to approach mobile security as a layered approach similar to the "defense in depth" approach that they embraced in their internal networks. This requires correlating and analyzing information from devices (whether iOS or Android), applications and network connections according to all the different threat vectors threat actors can use to ensure that nothing is missed.