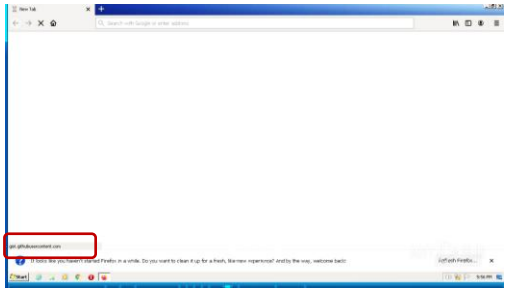
Incident Reponse Analysis

Scenario:

Lavoriamo in un'azienda in un SOC o CSIRT in una grande azienda e due utenti segnalano problemi sui loro computer e chiedono assistenza al reparto tecnico (che siamo noi)

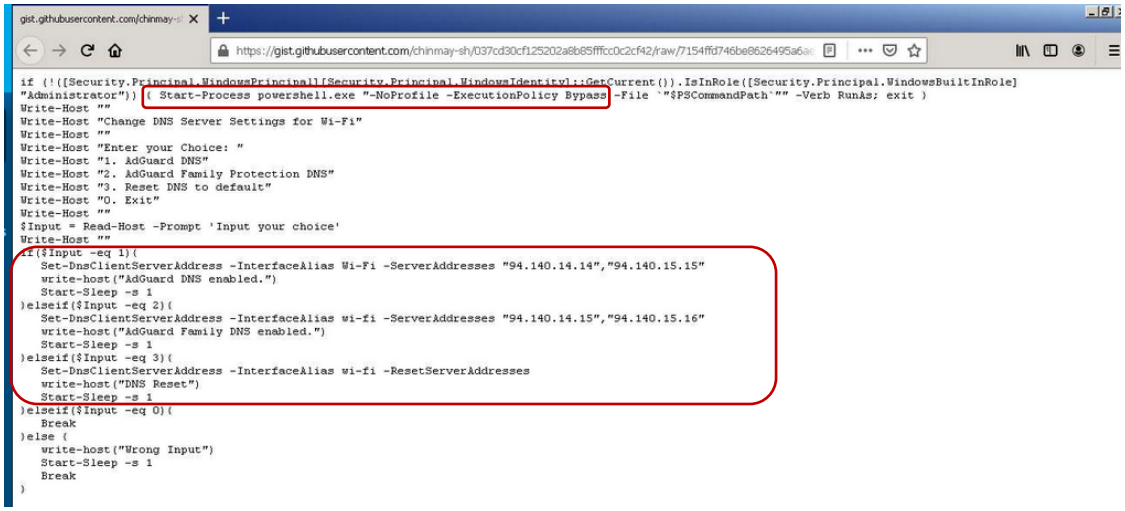
Analisi situazione https://tinyurl.com/linklosco1

L'utente ha aperto un sito web che contiene uno script Powershell:



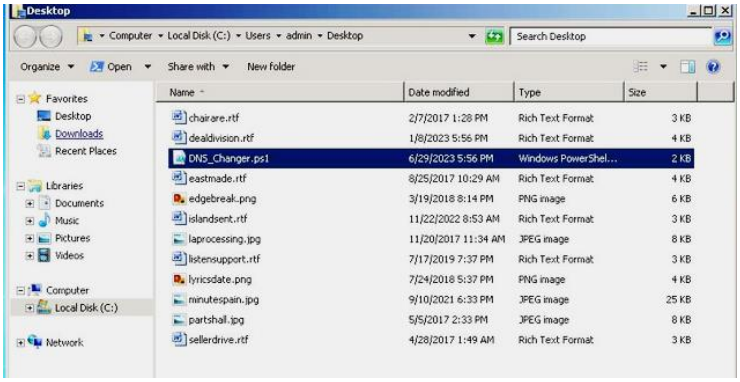
Uno sguardo rapido allo script ci mostra che lo script avvia il processo Powershell.exe con l'opzione `-ExecutionPolicy Bypass` che bypassa le politiche (conosciute come "Execution Policies") di Windows che determinano come gli script possono essere eseguiti. Queste politiche sono progettate per proteggere l'utente da script potenzialmente dannosi. Lo script viene eseguito quindi senza restrizioni ed è potenzialmente dannoso.

Vediamo più sotto alcuni pezzi di script che sembrano andare a modificare la configurazione DNS del dispositivo.

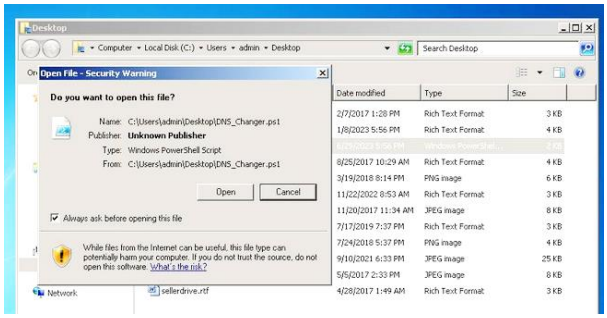


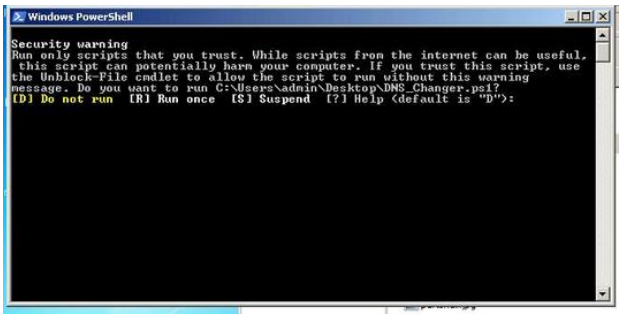
```
if (!([Security.Principal.WindowsPrincipal][Security.Principal.WindowsIdentity]::GetCurrent()).IsInRole([Security.Principal.WindowsBuiltInRole]
"Administrator")) { Start-Process powershell.exe "-NoProfile -ExecutionPolicy Bypass -File `"$PSCommandPath`"" -Verb RunAs; exit }
Write-Host ""
Write-Host "Change DNS Server Settings for Wi-Fi"
Write-Host ""
Write-Host "Enter your Choice: "
Write-Host "1. AdGuard DNS"
Write-Host "2. AdGuard Family Protection DNS"
Write-Host "3. Reset DNS to default"
Write-Host "0. Exit"
Write-Host ""
$Input = Read-Host -Prompt 'Input your choice'
Write-Host ""
if($Input -eq 1){
   Set-DnsClientServerAddress -InterfaceAlias Wi-Fi -ServerAddresses "94.140.14.14","94.140.15.15"
   write-host("AdGuard DNS enabled.")
   Start-Sleep -s 1
}elseif($Input -eq 2){
   Set-DnsClientServerAddress -InterfaceAlias wi-fi -ServerAddresses "94.140.14.15","94.140.15.16"
   write-host("AdGuard Family DNS enabled.")
   Start-Sleep -s 1
}elseif($Input -eq 3){
   Set-DnsClientServerAddress -InterfaceAlias wi-fi -ResetServerAddresses
   write-host("DNS Reset")
   Start-Sleep -s 1
}elseif($Input -eq 0){
   Break
}else {
   write-host("Wrong Input")
   Start-Sleep -s 1
   Break
}
```

Scaricando il file provando ad eseguirlo vediamo che il comportamento ipotizzato sembra proprio quello effettivo:



Un security warning ci avvisa che lo script può essere dannoso.





Andando avanti nell'esecuzione, vediamo appunto che lo script ci permette di modificare le impostazioni del server DNS :

L’analisi dei processi rileva le attività malevole e sospette generate dal processo powershell.exe, elencando tutti gli aspetti per cui sono considerate tali.

In sintesi, lo script viene eseguito senza restrizioni di sicurezza, ignorando tutte le impostazioni configurate, e legge (e permette di modificare) le impostazioni Internet e operando con account locali.

Behavior activities

MALICIOUS

Bypass execution policy to execute commands

- powershell.exe (PID: 3300)

SUSPICIOUS

Reads the Internet Settings

- powershell.exe (PID: 2272)
- powershell.exe (PID: 3300)

The process bypasses the loading of PowerShell profile settings

- powershell.exe (PID: 2272)

The process executes Powershell scripts

- powershell.exe (PID: 2272)

Application launched itself

- powershell.exe (PID: 2272)

Starts POWERSHELL.EXE for commands execution

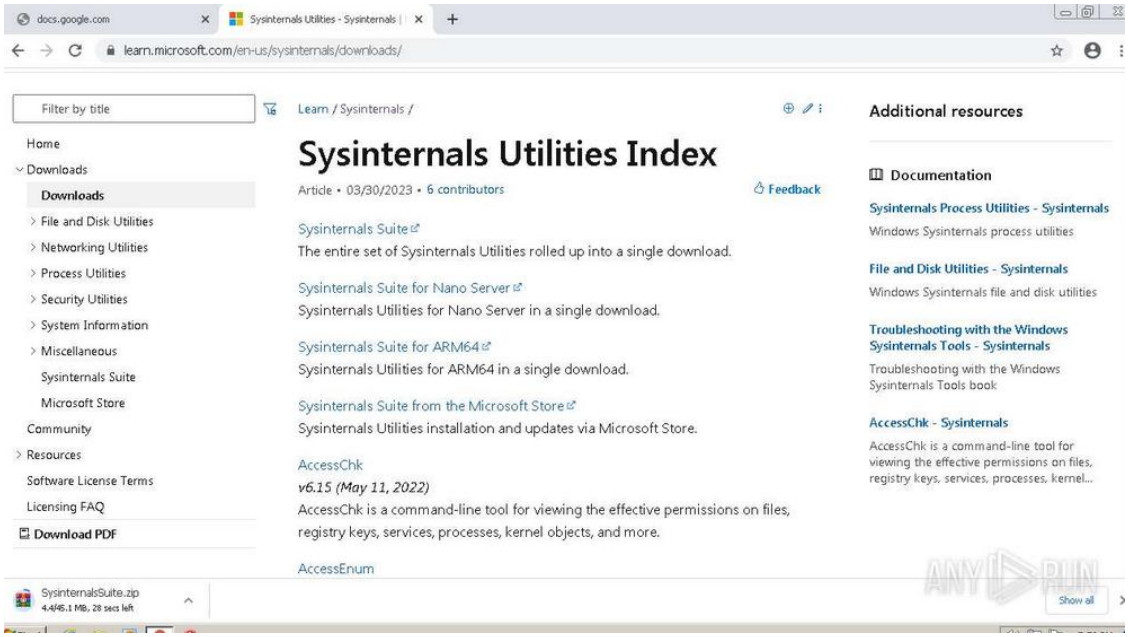
- powershell.exe (PID: 2272)

Using PowerShell to operate with local accounts

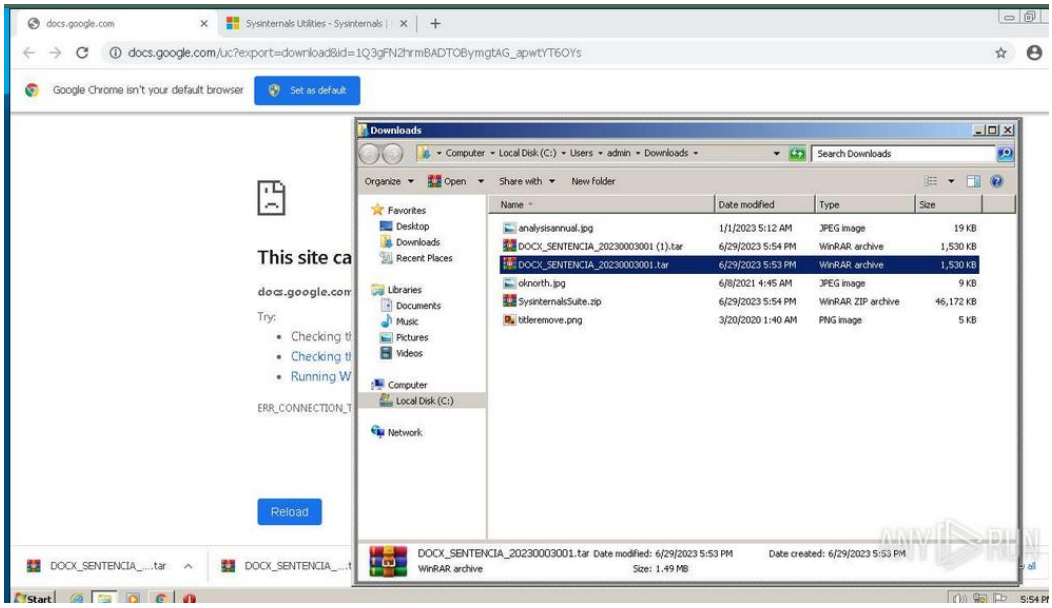
- powershell.exe (PID: 3300)

Analisi situazione https://tinyurl.com/linklosco2

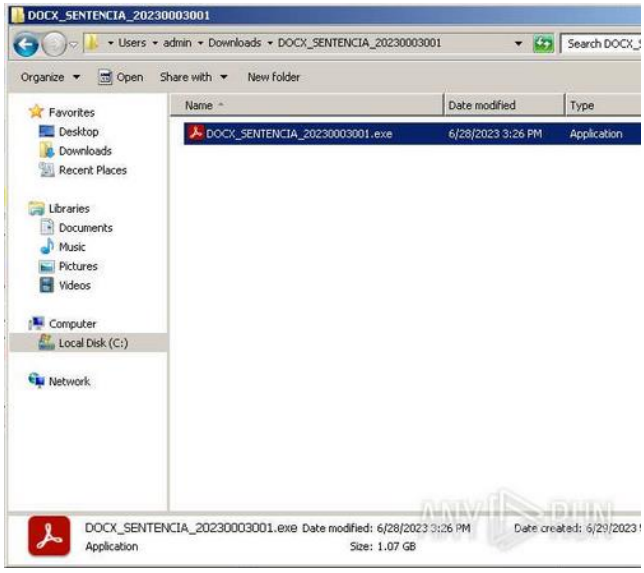
Da questo PC ccarichiamo Sysinternals, una suite di utilità sviluppata originariamente da Mark Russinovich e Bryce Cogswell e successivamente acquisita da Microsoft. Questa suite è molto popolare tra gli amministratori di sistema, i professionisti IT e gli esperti di sicurezza per la sua capacità di fornire approfondimenti dettagliati e funzionalità avanzate relative al sistema operativo Windows.



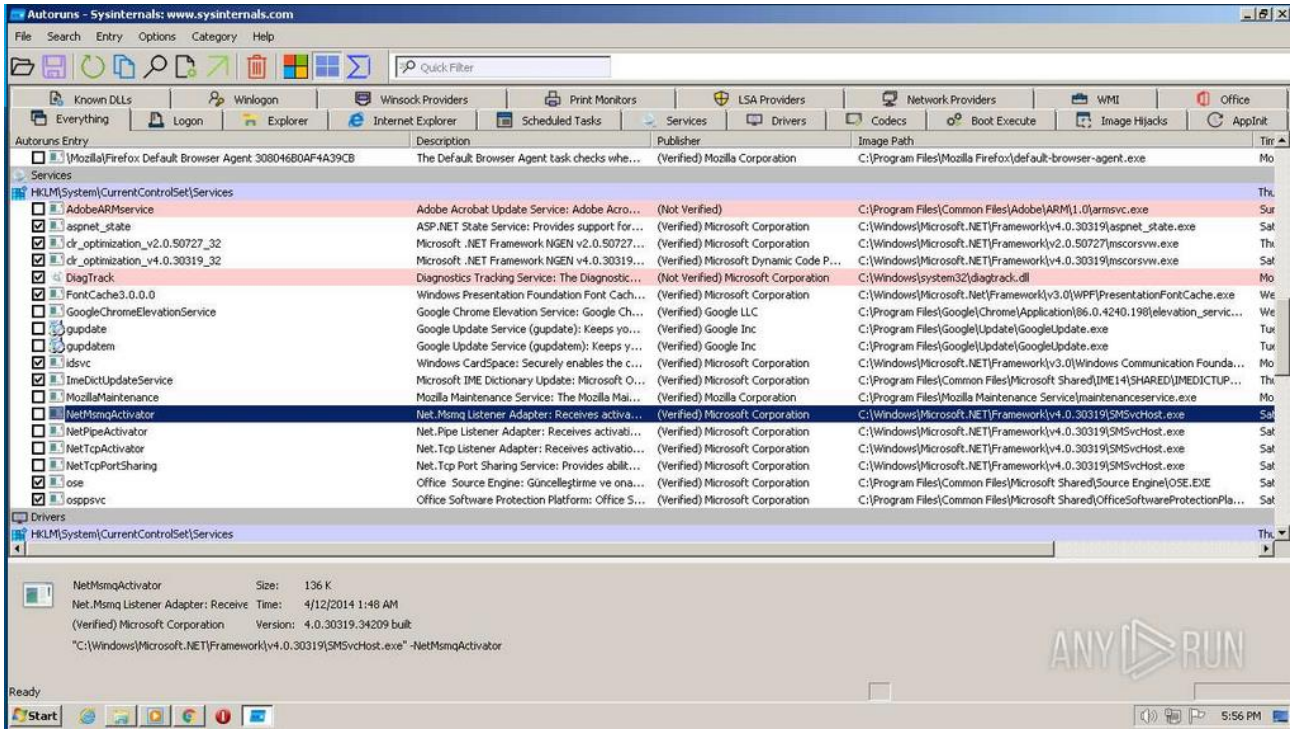
Con il download di Sysiternals, vediamo che vengono scaricati anche due file compressi .tar dal nome sospetto.



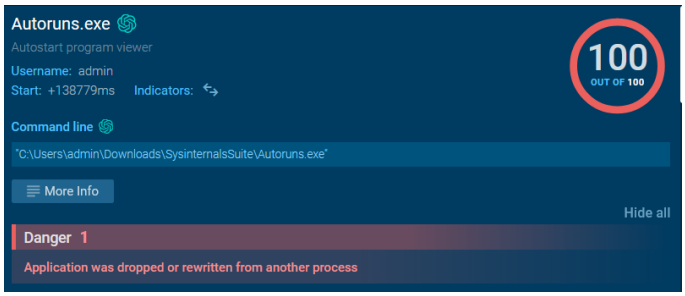
Estraendo i file scaricati dall'archivio, vediamo che ciascuno contiene un file eseguibile.



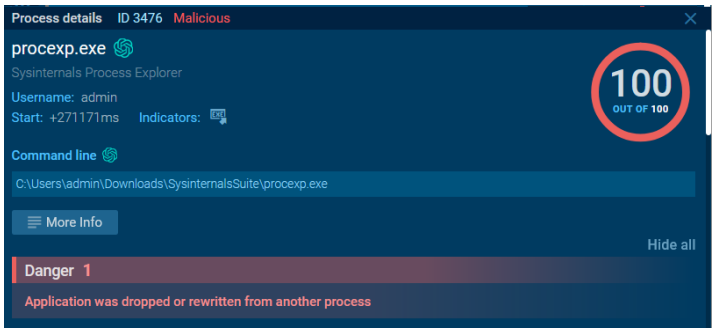
Eseguiamo Autoruns, un'utility progettata per mostrare quali programmi sono configurati per avviarsi durante il boot o l'accesso al sistema, e mostra le voci nell'ordine in cui Windows le elabora. Questi programmi includono quelli nel tuo avvio automatico, Run, RunOnce e altre chiavi del Registro di sistema.



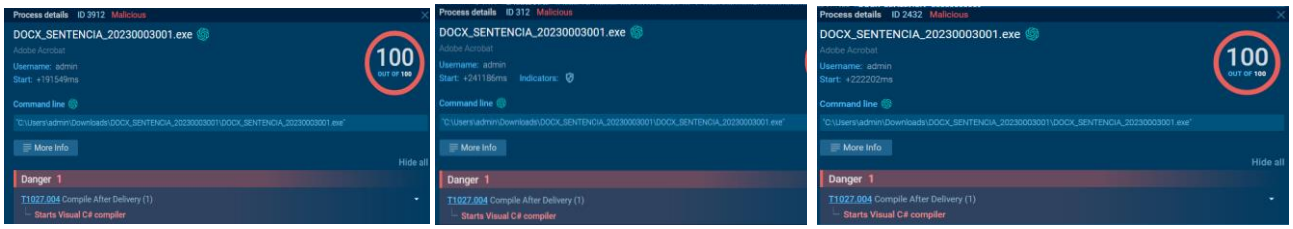
Un’analisi dei processi mostra che l’applicazione Autoruns è stata rimossa o sostituita da un altro processo.



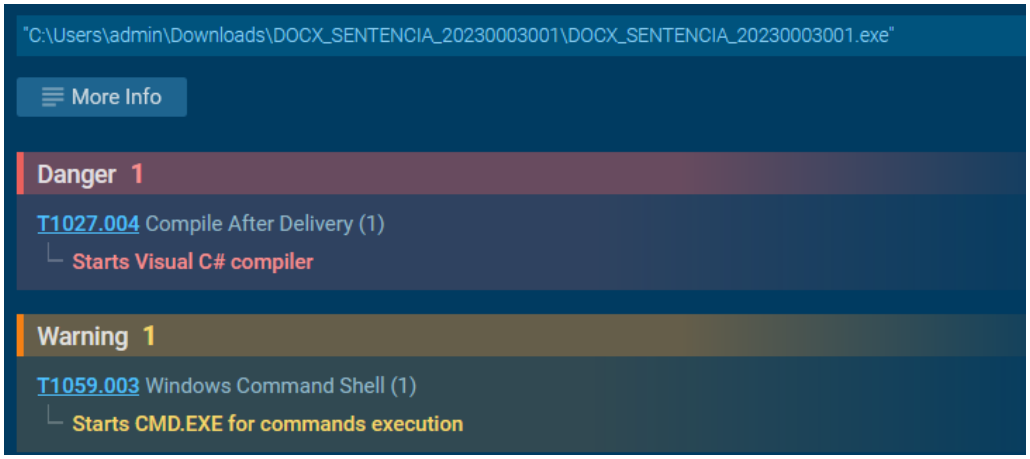
Lo stesso messaggio è riportato per l’applicazione procexp.exe, un’altra applicazione della suite Sysinternals.



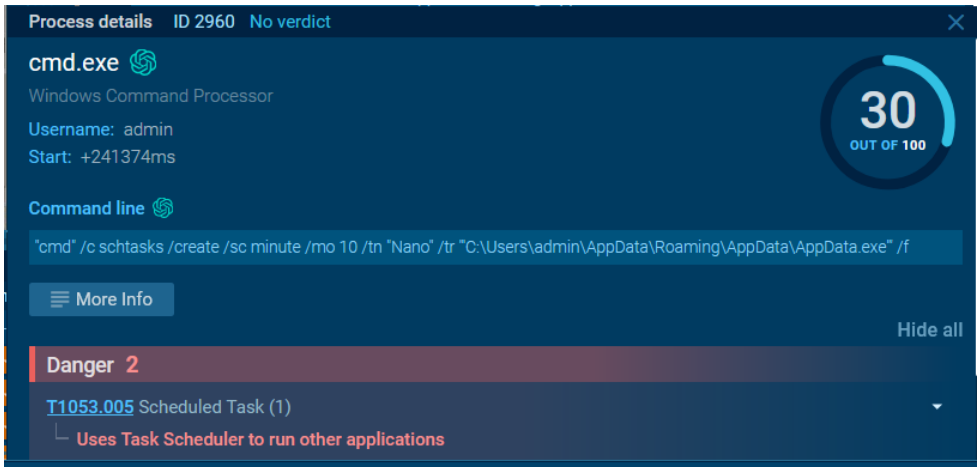
Si rileva attività malevola anche per gli eseguibili estratti dai file .tar sospetti scaricati da web. Entrambi i processi avviano il compilatore Visual C#.



Tutti questi processi avviano la shell dei comandi Windows cmd.exe per l’esecuzione di comandi

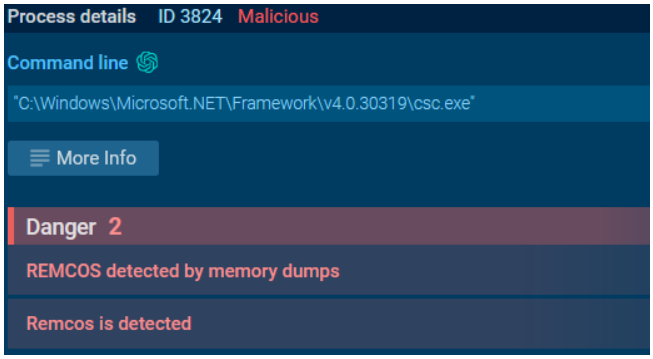


Il processo cmd.exe usa a sua volta il Task Scheduler per eseguire altri processi.



Sul processo csc.exe è stato rilevato il malware **Remcos.**

Remcos è un malware di tipo RAT che gli aggressori utilizzano per eseguire azioni su macchine infettate in remoto. Questo malware è estremamente attivo e viene costantemente aggiornato, con nuove versioni rilasciate quasi ogni mese.



Possiamo vedere in dettaglio che questo malware registra i tasti digitati dall'utente, legge le impostazioni internet e si connette a porte insolite.