TeamTNT, Group G0139 | MITRE ATT&CK®

Enterprise T1098 .004 Account Manipulation: SSH Authorized Keys

TeamTNT has added RSA keys in `authorized_keys`.[8][10]

Enterprise T1583 .001 Acquire Infrastructure: Domains

TeamTNT has obtained domains to host their payloads.[1]

Enterprise T1595 .001 Active Scanning: Scanning IP Blocks

TeamTNT has scanned specific lists of target IP addresses.[6]

.002 Active Scanning: Vulnerability Scanning

TeamTNT has scanned for vulnerabilities in IoT devices and other related resources such as the Docker API.[6]

Enterprise T1071 Application Layer Protocol

TeamTNT has used an IRC bot for C2 communications.[6]

.001 Web Protocols

TeamTNT has the `curl` command to send credentials over HTTP and the `curl` and `wget` commands to download new software.[3][4][10] TeamTNT has also used a custom user agent HTTP header in shell scripts.[6]

Enterprise T1547 .001 Boot or Logon Autostart Execution: Registry Run Keys / Startup Folder

TeamTNT has added batch scripts to the startup folder.[7]

Enterprise T1059 .001 Command and Scripting Interpreter: PowerShell

TeamTNT has executed PowerShell commands in batch scripts.[7]

.003 Command and Scripting Interpreter: Windows Command Shell

TeamTNT has used batch scripts to download tools and executing cryptocurrency miners.[7]

.004 Command and Scripting Interpreter: Unix Shell

TeamTNT has used shell scripts for execution.[6][10]

.009 Command and Scripting Interpreter: Cloud API

TeamTNT has leveraged AWS CLI to enumerate cloud environments with compromised credentials.[11]

.013 Command and Scripting Interpreter: Container CLI/API

TeamTNT targeted misconfigured containers and used container CLI tools.[12]

Enterprise T1609 Container Administration Command

TeamTNT executed Hildegard through the kubelet API run command and by executing commands on running containers.[5]

Enterprise T1613 Container and Resource Discovery

TeamTNT has checked for running containers with `docker ps` and for specific container names with `docker inspect`.[6] TeamTNT has also searched for Kubernetes pods running in a local network.[10]

Enterprise T1136 .001 Create Account: Local Account

TeamTNT has created local privileged users on victim machines.[3]

Enterprise T1543 .002 Create or Modify System Process: Systemd Service

TeamTNT has established persistence through the creation of a cryptocurrency mining system service using `systemctl`.[6][10]

.003 Create or Modify System Process: Windows Service

TeamTNT has used malware that adds cryptocurrency miners as a service.[7]

Enterprise T1074 .001 Data Staged: Local Data Staging

TeamTNT has aggregated collected credentials in text files before exfiltrating.[10]

Enterprise T1140 Deobfuscate/Decode Files or Information

TeamTNT has used a script that decodes a Base64-encoded version of WeaveWorks Scope.[10]

Enterprise T1610 Deploy Container

TeamTNT has deployed different types of containers into victim environments to facilitate execution.[3][6] TeamTNT has also transferred cryptocurrency mining software to Kubernetes clusters discovered within local IP address ranges.[10]

Enterprise T1587 .001 Develop Capabilities: Malware

TeamTNT has developed custom malware such as Hildegard.[5]

Enterprise T1611 Escape to Host

TeamTNT has deployed privileged containers that mount the filesystem of victim machine.[3][8]

Enterprise T1048 Exfiltration Over Alternative Protocol

TeamTNT has sent locally staged files with collected credentials to C2 servers using cURL.[10]

Enterprise T1133 External Remote Services

TeamTNT has used open-source tools such as Weave Scope to target exposed Docker API ports and gain initial access to victim environments.[3][10] TeamTNT has also targeted exposed kubelets for Kubernetes environments.[5]

Enterprise T1083 File and Directory Discovery

TeamTNT has used a script that checks `/proc/*/environ` for environment variables related to AWS.[10]

Enterprise T1222 .002 File and Directory Permissions Modification: Linux and Mac File and Directory Permissions Modification

TeamTNT has modified the permissions on binaries with `chattr`.[6][10]

Enterprise T1562 .001 Impair Defenses: Disable or Modify Tools

TeamTNT has disabled and uninstalled security tools such as Alibaba, Tencent, and BMC cloud monitoring agents on cloud-based infrastructure.[7][10]

.004 Impair Defenses: Disable or Modify System Firewall

TeamTNT has disabled `iptables`.[8]

Enterprise T1070 .002 Indicator Removal: Clear Linux or Mac System Logs

TeamTNT has removed system logs from `/var/log/syslog`.[8]

.003 Indicator Removal: Clear Command History

TeamTNT has cleared command history with `history -c`.[6][10]

.004 Indicator Removal: File Deletion

TeamTNT has used a payload that removes itself after running. TeamTNT also has deleted locally staged files for collecting credentials or scan results for local IP addresses after exfiltrating them.[7][10]

Enterprise T1105 Ingress Tool Transfer

TeamTNT has the `curl` and `wget` commands as well as batch scripts to download new tools.[3][10]

Enterprise T1680 Local Storage Discovery

TeamTNT has searched for disk partition and logical volume information.[7][10]

Enterprise T1036 Masquerading

TeamTNT has disguised their scripts with docker-related file names.[10]

.005 Match Legitimate Resource Name or Location

TeamTNT has replaced .dockerd and .dockerenv with their own scripts and cryptocurrency mining software.[10]

Enterprise T1046 Network Service Discovery

TeamTNT has used masscan to search for open Docker API ports and Kubernetes clusters.[4][5][10] TeamTNT has also used malware that utilizes zmap and zgrab to search for vulnerable services in cloud environments.[1]

Enterprise T1027 .002 Obfuscated Files or Information: Software Packing

TeamTNT has used UPX and Ezuri packer to pack its binaries.[6]

.013 Obfuscated Files or Information: Encrypted/Encoded File

TeamTNT has encrypted its binaries via AES and encoded files using Base64.[6][8]

Enterprise T1120 Peripheral Device Discovery

TeamTNT has searched for attached VGA devices using lspci.[10]

Enterprise T1057 Process Discovery

TeamTNT has searched for rival malware and removes it if found.[6] TeamTNT has also searched for running processes containing the strings aliyun or liyun to identify machines running Alibaba Cloud Security tools.[10]

Enterprise T1219 Remote Access Tools

TeamTNT has established tmate sessions for C2 communications.[5][10]

Enterprise T1021 .004 Remote Services: SSH

TeamTNT has used SSH to connect back to victim machines.[3] TeamTNT has also used SSH to transfer tools and payloads onto victim hosts and execute them.[10]

Enterprise T1496 .001 Resource Hijacking: Compute Hijacking

TeamTNT has deployed XMRig Docker images to mine cryptocurrency.[2][4] TeamTNT has also infected Docker containers and Kubernetes clusters with XMRig, and used RainbowMiner and lolMiner for mining cryptocurrency.[10]

Enterprise T1014 Rootkit

TeamTNT has used rootkits such as the open-source Diamorphine rootkit and their custom bots to hide cryptocurrency mining activities on the machine.[6] [10]

Enterprise T1518 .001 Software Discovery: Security Software Discovery

TeamTNT has searched for security products on infected machines.[7][10]

Enterprise T1608 .001 Stage Capabilities: Upload Malware

TeamTNT has uploaded backdoored Docker images to Docker Hub.[2]

Enterprise T1082 System Information Discovery

TeamTNT has searched for system version, architecture, and hostname information.[7][10]

Enterprise T1016 System Network Configuration Discovery

TeamTNT has enumerated the host machine’s IP address.[6]

Enterprise T1049 System Network Connections Discovery

TeamTNT has run `netstat -anp` to search for rival malware connections.[6] TeamTNT has also used `libprocesshider` to modify `/etc/ld.so.preload`.[7]

Enterprise T1007 System Service Discovery

TeamTNT has searched for services such as Alibaba Cloud Security's aliyun service and BMC Helix Cloud Security's bmc-agent service in order to disable them.[10]

Enterprise T1569 .003 System Services: Systemctl

TeamTNT has created system services to execute cryptocurrency mining software.[10]

Enterprise T1552 .001 Unsecured Credentials: Credentials In Files

TeamTNT has searched for unsecured AWS credentials and Docker API credentials.[4][6][10]

.004 Unsecured Credentials: Private Keys

TeamTNT has searched for unsecured SSH keys.[4][6]

.005 Unsecured Credentials: Cloud Instance Metadata API

TeamTNT has queried the AWS instance metadata service for credentials.[6][10]

Enterprise T1204 .003 User Execution: Malicious Image

TeamTNT has relied on users to download and execute malicious Docker images.[2]

Enterprise T1102 Web Service

TeamTNT has leveraged iplogger.org to send collected data back to C2.[8][10]
