Your IT Company

Principais Vulnerabilidades e Ameaças (abril/26)

Sumário

1. Objetivo

Este documento foi desenvolvido e fundamentado nas documentações oficiais do NIST (National Institute Of Standards and Technology), MITRE e CVE.org, que são mundialmente conhecidas na área de cibersegurança como modelos de padronização das melhores práticas de segurança, pesquisa de metodologias de ataques, defesa cibernética e catalogação de novas vulnerabilidades em sistemas operacionais e aplicações.

2. Vulnerabilidades e Ameaças descobertas

2.1. Chrome zero-day explorado ativamente (CVE-2026-5281)

Foi divulgada uma vulnerabilidade zero-day ativa em Google Chrome (registrada como CVE-2026-5281) que permite execução remota de código ao explorar um bug do tipo Use-After-Free no motor WebGPU (Google Dawn).

A falha foi adicionada ao catálogo Known Exploited Vulnerabilities (KEV) em 1º de abril de 2026 e exige atualização imediata dos navegadores para reduzir risco de comprometimento.

Exploração

O A falha é um Use-After-Free no **Google Dawn**, implementação open-source do WebGPU utilizada na renderização gráfica em navegadores baseados em Chromium. Esse tipo de vulnerabilidade ocorre quando o sistema continua utilizando um ponteiro para memória já liberada, possibilitando corrupção de memória e potencial execução de código.

O vetor de exploração envolve um atacante remoto que induz a vítima a acessar uma página HTML especialmente criada. Ao processar esse conteúdo, o navegador pode acionar a condição de UAF no processo de renderização, permitindo a execução de código arbitrário no contexto do sistema da vítima.

Como o componente afetado faz parte do ecossistema Chromium, o impacto se estende a diversos navegadores como **Google Chrome**, **Microsoft Edge**, **Brave**, **Opera** e **Vivaldi**, até que correções sejam aplicadas por cada fornecedor.

O impacto potencial inclui comprometimento total do endpoint, roubo de dados sensíveis, instalação silenciosa de malware e movimentação lateral em ambientes corporativos.

Mitigação e Prevenção

- **Ações imediatas e recomendações práticas:**
  - Atualizar imediatamente: aplique o patch do seu fornecedor de navegador assim que disponível. Priorize essa correção nos ciclos de patch management.
- **Para administradores de TI / segurança:**
  - Forçar atualização automática e bloquear versões vulneráveis via políticas (GPO, MDM, SCCM, etc.).
  - Monitorar e aplicar atualizações também a navegadores Chromium-based usados na organização (Edge, Opera, Vivaldi, Brave).
  - Segmentar redes e limitar privilégios de endpoints para reduzir movimentação lateral em caso de comprometimento.

- Considerar políticas de navegação restrita (bloquear sites não confiáveis) e filtros web/proxys para reduzir exposição a páginas maliciosas.

- **Para administradores de TI / segurança:**
  - Reinicie e atualize o navegador assim que a atualização estiver disponível.
  - Evite visitar links ou páginas de origem desconhecida e não clique em anexos ou links suspeitos.
  - Controles adicionais: assinatura das atualizações do CISA KEV e feeds de vulnerabilidades para priorização de remediação; revisar logs e alertas relacionados a renderer crashes e conexões a URLs suspeitas.
  - Medida temporária extrema: se não for possível aplicar mitigação/patch, considere descontinuar o uso do produto vulnerável até que haja correção.

Para saber informações mais detalhadas sobre esta vulnerabilidade acesse: https://cybersecuritynews.com/chrome-0-day-flaw-exploited/.

2.2. Falha crítica de SQLi no FortiClient EMS (7.4.4) está sendo explorada ativamente

FortiClient Endpoint Management Server (EMS) versão 7.4.4 apresenta uma vulnerabilidade crítica de SQL injection que já está sendo explorada em ataques reais, permitindo execução remota de comandos sem autenticação. A falha oferece um vetor de acesso inicial atraente para atores de ameaça, com potencial para roubo de dados, implantação de malware e movimento lateral em redes corporativas.

Exploração

A vulnerabilidade é uma SQL Injection explorável via cabeçalho HTTP Site no painel administrativo web do FortiClient EMS. A falha ocorre por falta de sanitização adequada de entradas fornecidas pelo cliente.

O vetor de ataque consiste no envio de requisições HTTP GET maliciosas, onde o cabeçalho Site contém payloads SQL, como x'; SELECT pg_sleep(4)--, direcionadas a endpoints como /api/v1/init_consts. Como não há validação apropriada, o servidor processa essas entradas, permitindo a execução de comandos SQL arbitrários.

A exploração não requer autenticação, o que aumenta significativamente a criticidade. Um atacante pode alcançar execução remota de comandos e potencial comprometimento total do servidor EMS. Os impactos incluem exfiltração de dados sensíveis, implantação de malware ou ransomware e movimentação lateral na rede.

A exposição é relevante, com cerca de mil instâncias acessíveis publicamente segundo levantamentos em mecanismos como **Shodan**. Também foi observada campanha ativa explorando a falha, com tráfego associado ao IP 104.192.92.135.

A severidade foi classificada como CVSS 9.1 pela **Fortinet**. A vulnerabilidade foi descoberta por Gwendal Guégniaud e divulgada em 06/02/2026. A versão afetada é a 7.4.4, enquanto as versões 7.2, 8.0 e a edição FortiEMS Cloud não são impactadas.

Mitigação e Prevenção

- **Ações imediatas:**
  - **Atualizar**: aplicar imediatamente o patch oficial — atualizar FortiClient EMS 7.4.4 para 7.4.5 (mitigação definitiva).
  - **Inventário e exposição**: inventariar todas as instâncias EMS, identificar aquelas expostas à Internet e minimizar o acesso público (bloquear/filtrar).
  - **Restrição de acesso**: limitar acesso à interface administrativa via VPN, listas de controle de acesso (ACLs) ou IP allowlists; evitar exposição direta à Internet.
  - **Monitoramento e detecção**: inspecionar logs de proxy/WAF/reverse proxy e servidores web por requisições GET ao endpoint administrativo (/api/v1/init_consts e similares) com cabeçalho Site contendo caracteres incomuns ou SQL (palavras-chave como SELECT, pg_sleep, UNION, --, ';). Implementar regras IDS/IPS/WAF que detectem/mitiguem injeção via cabeçalhos HTTP.
  - **Resposta a incidentes**: em caso de detecção, isolar o host comprometido, coletar evidências (logs, memória), realizar varredura por persistência e cargas secundárias, e executar procedimento de contenção e recuperação (reinstalar a partir de imagens limpas se necessário).
- **Administradores de sistema:**
  - Priorizar atualização para 7.4.5, bloquear acessos administrativos públicos, revisar logs e configurar regras WAF/IDS.

IOCs

- **IP observado como origem de ataque:** 104.192.92.135
- **Endpoint alvo observado:** /api/v1/init_consts
- **Exemplo de cabeçalho injetado observado:** Site: x'; SELECT pg_sleep(4)—
- **Padrão de requisição suspeita:** HTTP GET para interface administrativa com caracteres SQL no cabeçalho Site (palavras-chave: SELECT, pg_sleep, --, ';)

Para saber informações mais detalhadas sobre esta vulnerabilidade acesse: https://cybersecuritynews.com/forticlient-ems-vulnerability-exploited-/.

2.3. BlueHammer PoC para Windows Defender: PoC público explora corrida TOCTOU na atualização de assinaturas

BlueHammer é um PoC público que explora uma vulnerabilidade zero-day de elevação de privilégio local (LPE) no mecanismo de atualização de assinaturas do Microsoft Defender. A técnica combina uma condição de corrida TOCTOU com confusão de caminho para forçar a cópia do hive SAM para um local acessível, permitindo extração de hashes e tentativa de escalonamento para privilégios elevados — risco imediato para ambientes com contas locais ativas.

Exploração

O vetor de ataque explora a interface RPC interna do **Microsoft Defender Antivirus**, especificamente o serviço IMpService e a chamada ServerMpUpdateEngineSignature durante o processamento de arquivos de atualização como mpasbase.vdm.

O mecanismo técnico se baseia em uma condição de corrida do tipo TOCTOU. O exploit aguarda a disponibilização de uma atualização via Windows Update, baixa o arquivo legítimo e aplica um *oplock* (opportunistic lock) sobre ele. Esse bloqueio permite interceptar o acesso privilegiado do Defender no momento exato em que o arquivo é manipulado.

Quando o *oplock* é acionado, o atacante move o arquivo legítimo, recria o diretório como um *reparse point* e insere um link simbólico no Object Manager que redireciona a leitura do processo — executando como NT AUTHORITY\SYSTEM — para um caminho baseado em VSS do hive SAM. Como resultado, o Defender acaba copiando o arquivo SAM para o diretório temporário do sistema.

Na fase de pós-exploração, o conteúdo do SAM é processado para extrair hashes NTLM, em abordagem similar a ferramentas como **Mimikatz**. Caso uma conta administradora válida seja identificada, o exploit tenta redefinir temporariamente a senha, autenticar via LogonUserEx e criar ou executar um serviço para obter execução com privilégios elevados.

A técnica possui limitações práticas, pois depende do timing das atualizações do Defender, da disponibilidade dos pacotes e do estado das contas locais. Testes indicaram falhas em cenários específicos, como contas desabilitadas, e em ambientes Windows Server o resultado pode se limitar à elevação para administrador, sem necessariamente alcançar SYSTEM. O PoC foi divulgado publicamente e, até o momento, não há correção oficial disponibilizada.

Mitigação e Prevenção

- **Para administradores de TI / SOC:**
  - Monitorar eventos relacionados a acessos e criação de links simbólicos e reparse points nas pastas de atualização do Defender (ex.: alertar em criação de reparse points sob C:\ProgramData\Microsoft\Windows Defender\Definition Updates).
  - Configurar detecções para o Event ID 4663 (acesso a objetos) em diretórios do Defender e criar alertas para padrões anômalos.
  - Detectar acessos VSS (Volume Shadow Copy) seguidos de gravações anômalas em %TEMP% que possam representar vazamento do hive SAM.
  - Implementar regras de EDR/IDS para identificar leituras privilegiadas que resolvam por symlinks do Object Manager.

- Revisar e endurecer permissões NTFS e ACLs nas pastas de atualização do Defender e %TEMP%.
- Auditar e desabilitar contas locais desnecessárias; aplicar políticas que bloqueiem contas locais com privilégios e usar gerenciamento centralizado (ex.: Active Directory, LAPS) para senhas locais.
- Verificar serviços e contas criadas recentemente e investigar tentativas de criação/arranque de serviços suspeitos.

- **Para administradores de endpoint:**
  - Habilitar proteção contra manipulação (Tamper Protection) e assegurar políticas de atualização centralizadas para Defender.
  - Atualizar e reforçar regras de EDR para bloqueio e detecção de técnicas de reparse point / oplock exploitation.
- **Para usuários finais:**
  - Evitar manter contas locais administrativas habilitadas em estações de trabalho; usar contas com privilégio mínimo.
  - Reportar qualquer comportamento estranho do sistema (erros de atualização do Defender, arquivos inesperados em %TEMP%).

IOCs

- **Nome do arquivo alvo:** mpasbase.vdm
- **Caminho/ diretório de atualização:** C:\ProgramData\Microsoft\Windows Defender\Definition Updates
- **Object Manager symlink observado:** \BaseNamedObjects\Restricted\mpasbase.vdm
- **Hive alvo e caminho VSS usado:** \Windows\System32\Config\SAM
- **Local de exfiltração temporária observado:** %TEMP% (arquivos com conteúdo de hive SAM)
- **Event ID associado:** 4663 (acessos a objetos)
- Reparse point / criação de symlink inespera sob a pasta de definições do Defender
- **Nome do Cloud Files provider embutido no PoC:** IHATEMICROSOFT
- **String de senha hardcoded encontrada no PoC:** $PWNed666!!!WDFAIL
- **Observável comportamental:** acesso a VSS seguido de gravação em %TEMP% + leituras privilegiadas resolvidas por object manager symlinks

Para saber informações mais detalhadas sobre esta vulnerabilidade acesse: https://cybersecuritynews.com/bluehammer-poc-for-windows-defender/.

2.4. 50.000 sites WordPress expostos por falha crítica no plugin “Ninja Forms – File Upload

A falha crítica no plugin Ninja Forms File Upload (CVE-2026-0740) permite upload arbitrário de arquivos sem autenticação, possibilitando a execução remota de código (RCE) e tomada total de sites. Cerca de 50.000 instalações foram apontadas como potencialmente vulneráveis, tornando a atualização imediata essencial para administradores.

Exploração

A vulnerabilidade está no processamento de uploads do add-on, especificamente nas funções handle_upload() e _process(), que utilizam move_uploaded_file() para mover arquivos temporários sem validações adequadas.

O problema ocorre porque, embora o plugin valide o tipo do arquivo enviado, ele não valida corretamente a extensão do arquivo final nem sanitiza o nome do arquivo. Essa falha permite manipulação do caminho e exploração de path traversal, possibilitando que um atacante envie arquivos maliciosos, como .php, diretamente para a raiz do site.

Com isso, é possível carregar um webshell e executá-lo no servidor, resultando em execução remota de código. A partir desse ponto, o invasor pode exfiltrar bancos de dados, injetar malware, redirecionar tráfego ou utilizar o servidor comprometido para ampliar ataques.

Todas as versões até 3.3.26 são afetadas. Uma correção parcial foi introduzida na versão 3.3.25, e a correção completa foi disponibilizada na versão 3.3.27, lançada em 19/03/2026. Regras de mitigação também foram disponibilizadas pela **Wordfence** em janeiro e fevereiro de 2026, ajudando a bloquear tentativas de exploração.

Mitigação e Prevenção

- **Ações imediatas:**
  - Atualize o plugin Ninja Forms — File Upload para a versão *3.3.27* ou superior imediatamente.
  - Se não puder atualizar de imediato, desative o add-on até a aplicação da correção.
  - Aplique regras de WAF (Web Application Firewall) para bloquear uploads .php e strings de path traversal; habilite assinaturas específicas (ex.: Wordfence ou outro WAF).
  - Revise permissões de diretórios de upload (evite que uploads sejam gravados na raiz do site; use diretórios com permissão mínima e bloqueio de execução).
  - Use .htaccess/nginx config para bloquear execução de arquivos PHP nas pastas de upload (ex.: deny access a *.php nesses diretórios).
  - Faça varredura imediata por webshells e arquivos suspeitos nas pastas de uploads e na raiz do site; procure por arquivos .php recentes e por nomes com caracteres estranhos.

- Caso haja suspeita de comprometimento: isole o site, faça backup para investigação, revise logs (acessos e erros), troque senhas de contas administrativas e credenciais de banco, e considere restaurar de backup limpo ou recriar o ambiente.
- Verifique integridade dos temas, plugins e core do WordPress; atualize tudo para versões suportadas.
- Habilite monitoramento contínuo e alertas para uploads anômalos e execuções de scripts.

- **Recomendações para usuários finais / proprietários de sites sem equipe técnica:**
  - Entre em contato com o provedor de hospedagem ou responsável técnico e peça atualização imediata do plugin.
  - Caso perceba comportamento estranho (redirecionamentos, páginas desconhecidas, defacement), leve o site offline temporariamente até verificação.
  - Faça backup completo antes de qualquer alteração e mantenha cópias offsite.

Para saber informações mais detalhadas sobre esta vulnerabilidade acesse: https://cybersecuritynews.com/50000-wordpress-sites-exposed/.

2.5. Vulnerabilidade de Injeção de Comandos no OpenAI Codex permite roubo de tokens do GitHub

OpenAI Codex apresenta uma vulnerabilidade crítica de injeção de comandos que permite a exfiltração de tokens de acesso do GitHub. A falha ocorre ao passar diretamente o nome da branch para scripts de configuração do container, possibilitando execução de payloads e movimento lateral em ambientes de desenvolvimento com impacto em web, CLI, SDK e extensões IDE do Codex.

Exploração

O OpenAI Codex opera tarefas de geração de código e análise de repositórios por meio de containers gerenciados, ativados sob demanda pelas requisições dos usuários. No fluxo de configuração desses containers, o parâmetro de nome da branch é processado por scripts de setup sem a devida sanitização. Essa falha de validação abre caminho para a injeção de comandos shell, permitindo que instruções arbitrárias sejam executadas no ambiente de inicialização.

A exploração dessa brecha utiliza nomes de branch contendo payloads maliciosos para forçar a exportação do token OAuth do GitHub para arquivos de texto legíveis. O atacante manipula o agente Codex para ler esses arquivos e exibir o segredo diretamente na interface web. Para evadir sistemas de detecção e restrições de nomenclatura, utilizam-se técnicas de ofuscação, como o uso de caracteres Unicode Ideographic Space e a substituição de espaços por separadores internos, tornando o ataque visualmente imperceptível.

O risco se estende ao ambiente local dos desenvolvedores, onde o arquivo auth.json armazena tokens de sessão em clientes desktop. Com o acesso físico ou remoto à máquina da vítima, um invasor utiliza esses tokens para autenticação na API de backend, permitindo a extração de históricos de tarefas e a recuperação de tokens sensíveis expostos nos logs dos containers.

A vulnerabilidade apresenta um alto potencial de automação e replicação em cadeia. Ao introduzir uma branch infectada em repositórios compartilhados, o ataque compromete qualquer usuário ou sistema que interaja com o código, incluindo automações de Pull Requests (PRs) que disparam containers de revisão. Esse cenário resulta no roubo sistemático de Installation Access Tokens. O problema afeta o ecossistema integrado do Codex incluindo o site do ChatGPT, CLI, SDK e extensões de IDE com o ciclo de remediação estabelecido entre o reporte em dezembro de 2025 e a correção definitiva em janeiro de 2026.

Mitigação e Prevenção

- **Recomendações Gerais:**
  - **Sanitização:** sanitizar rigorosamente todos os inputs controláveis por usuários antes de passá-los a comandos shell ou scripts de inicialização.
  - **Isolamento de containers:** tratar containers de agentes como limites de segurança fortes rodar com privilégios mínimos, filesystem imutável e sem credenciais em variáveis de ambiente ou arquivos acessíveis sem necessidade.
  - **Princípio do menor privilégio:** auditar e reduzir permissões concedidas a integrações/agents (tokens com escopo mínimo e tempo de vida curto).
  - **Rotação de tokens:** rotacionar tokens do GitHub regularmente e revogar imediatamente tokens suspeitos.
  - **Monitoramento:** monitorar repositórios por nomes de branch incomuns (metacaracteres de shell, caracteres Unicode ideográficos) e por criação de branches/PRs automatizadas inesperadas.
  - **Proteção local:** proteger credenciais locais (ex.: auth.json) com permissões restritas, criptografia quando possível e políticas de EDR/antivírus para detectar exfiltração.
  - **Validação de fontes externas:** não confiar em formatos ou entradas vindas de provedores externos sem validação adicional.
  - **Logs e auditoria:** habilitar logs de API e auditoria, revisar atividades incomuns (acessos fora do horário, requisições de token, leituras de arquivos sensíveis).
  - **Hardening CI/CD:** desabilitar execução automática de containers para revisões sem validação; limitar bots a ambientes isolados; bloquear execução de comandos shell a partir de nomes de branch/PR.
- **Recomendações para públicos específicos:**
  - **Administradores de sistema / segurança:** aplicar patches distribuídos pela OpenAI, revisar políticas de IAM, segregar ambientes e configurar alertas para criação de branches com caracteres suspeitos.

- **Equipes de desenvolvimento:** evitar armazenar tokens com escopo amplo localmente; usar credential managers e MFA; validar nomes de branches e revisar automatizações que acionam containers.
- **Usuários finais / desenvolvedores:** atualizar clientes Codex/ChatGPT, restringir permissões ao autorizar conectores GitHub e reportar branches suspeitas no repositório.

IOCs

- **Arquivo local que armazenou tokens:** *auth.json*
- **Padrão de nomes de branch suspeitos:** branches contendo Unicode Ideographic Space (U+3000) ou metacaracteres de shell ocultos e uso de separadores internos para burlar restrições de nome.

Para saber informações mais detalhadas sobre esta vulnerabilidade acesse: https://cybersecuritynews.com/openai-codex-command-injection-vulnerability/.

Esta publicação tem como objetivo orientar a equipe interna e os nossos clientes sobre ameaças emergentes no ambiente cibernético, possibilitando assim ações proativas de prevenção para tornar a infraestrutura de nossas empresas cada vez mais segura.

A publicação do documento de Inteligência de Ameaças será realizada quinzenalmente.

Produzido por: Equipe de Threat Intelligence da Service IT Security