Micro Focus

Conteúdo de segurança do software Fortify

Atualização 3 de 2020

sexta-feira, 25 de setembro de 2020

Sobre o Micro Focus Fortify Software Security Research

A equipe do Fortify Software Security Research converte pesquisa de ponta em inteligência de segurança que fortalece o portfólio de produtos Fortify, incluindo o Fortify Static Code Analyzer (SCA), o Fortify WebInspect e o Fortify Application Defender. Atualmente, o Conteúdo de segurança do software Micro Focus Fortify oferece suporte a 1.032 categorias de vulnerabilidade em 27 linguagens de programação e se estende por mais de um milhão de APIs individuais.

Saiba mais em: https://software.microfocus.com/en-us/software/security-research

O Fortify Software Security Research (SSR) tem a satisfação de anunciar a disponibilidade imediata de atualizações para o Fortify Secure Coding Rulepacks (versão em inglês 2020.2.0), Fortify WebInspect SecureBase (disponível via SmartUpdate) e Fortify Premium Content.

Micro Focus Fortify Secure Coding Rulepacks [SCA]

Nesta versão, os Pacotes de regras de codificação segura do Fortify detectam 815 categorias únicas de vulnerabilidades em 27 linguagens de programação e abrangem mais de um milhão de APIs individuais. Em resumo, essa versão inclui o seguinte:

Suporte .NET System.Text.Json (versão 3.1)[1]

O namespace System.Text.Json é uma alternativa de alta velocidade, compatível com os padrões e integrada ao pacote Json.NET da Newtonsoft. O namespace System.Text.Json também contém tipos para ler e gravar texto JSON codificado como UTF-8. O suporte inclui fluxo de dados por meio de vários métodos de serialização/desserialização, bem como maior cobertura para as seguintes categorias:

- Cross-Site Scripting: Reflected
- Cross-Site Scripting: Persistent
- Cross-Site Scripting: Poor Validation
- Privacy Violation
- Vazamento de informações do sistema: External
- Vazamento de informações do sistema: interno

Corrotinas e suporte para biblioteca padrão Kotlin[2]

Cobertura estendida das bibliotecas padrão Kotlin, incluindo suporte para corrotinas.
As corrotinas fornecem uma API sofisticada e flexível para desenvolver código simultâneo e assíncrono. Esta atualização acrescenta uma análise de fluxo de dados aprimorada em aplicativos que usam corrotinas.

Interoperabilidade Kotlin Java e aprimoramentos do Android[3]

Suporte de linguagem aprimorado em categorias de vulnerabilidades existentes para projetos que contêm código-fonte Java e Kotlin, bem como para aplicativos Android desenvolvidos usando Kotlin.

Aprimoramentos do Java 14[4]

Melhorias nas categorias de vulnerabilidades existentes para oferecer suporte às mudanças de versão até o Java 14.

Atualizações da versão do ECMAScript[5]

Atualizada a cobertura das regras das versões do ECMAScript para prover suporte a todas as versões até o ECMAScript 2020.

---

[1] Devido ao design do namespace System.Text.Json, ele não está vulnerável aos mesmos problemas de injeção de JSON relatados no pacote Newtonsoft Json.NET.
[2] APIs de corrotinas experimentais, como aquelas que usam expressões selecionadas, não têm suporte nesta atualização.
[3] Requer SCA 20.2 ou posterior.
[4] Requer SCA 20.2 ou posterior.
[5] Para um suporte completo, requer SCA 20.2 ou posterior.

Go Logrus (Versão 1.6.0)

Logrus é um logger estruturado para Go, totalmente compatível com a API do logger de biblioteca padrão. Logrus é um pacote de registro (logging) comum, que pode apresentar riscos relacionados ao registro para aplicativos corporativos. As categorias com suporte incluem:

- Prática deficiente de registro em log: Uso de um fluxo de saída do sistema
- Privacy Violation
- Vazamento de informações do sistema

Suporte ao Dockerfile[6]

O Docker permite que o código de empacotamento e suas dependências num ambiente de sandbox (também chamado de contêineres) sejam executados em qualquer ambiente de informática. Dockerfile é a configuração de imagens de contêiner. Esta atualização lida com as vulnerabilidades mais comuns do Dockerfile.

O suporte inicial cobre a categoria “Password Management: Password in Configuration File” que já existe, bem como as seguintes categorias novas:

- Dockerfile Misconfiguration: Default User Privilege
- Dockerfile Misconfiguration: Privileged Container
- Dockerfile Misconfiguration: Privileged Port
- Dockerfile Misconfiguration: Sensitive Host Discovery
- Dockerfile Misconfiguration: SSH Service

DISA STIG 4.11

Para dar suporte a nossos clientes federais na área de conformidade, foi adicionada uma correlação do Micro Focus Fortify Taxonomy ao Application Security e ao Development STIG, versão 4.11, da DISA (Defense Information Systems Agency).

Erratas diversas

Nesta versão, continuamos a investir recursos para garantir que possamos reduzir o número de problemas de falsos positivos e melhorar a capacidade dos clientes de auditar problemas. Os clientes também podem esperar alterações nos problemas relatados relacionadas ao seguinte:

Aviso de suspensão de uso de versões SCA anteriores a 17.x:
Avisamos aos nossos usuários de que esta será a última versão dos Rulepacks compatíveis com versões SCA anteriores a 17.x. No próximo lançamento, as versões SCA anteriores a 17.x não vão mais carregar os Rulepacks. Será necessário fazer o downgrade dos rulepacks ou o upgrade da versão do SCA.
Nas versões futuras, vamos continuar a oferecer suporte para as últimas quatro versões principais do SCA.

[6] Requer SCA 20.2 ou posterior.

Mapeamentos obsoletos – Guia de implementação técnica de segurança versões 3.x (STIG 3.x):
A equipe de Pesquisa de Segurança de Software continuou criando mapeamentos de categorias do Fortify para padrões externos e práticas recomendadas ao longo dos anos. Com isso, chegamos ao ponto em que as versões de diferentes padrões com suporte ficaram muito grandes, sendo que muitas versões antigas não são mais usadas.
Assim, agora marcamos as versões do STIG anteriores à versão 4.0 como “obsoletas” usando um novo atributo XML no arquivo externalmetadata.xml. Essa mudança significa que os aplicativos Fortify Software Security Center e SCA, a partir da versão 20.2, não exibirão mais essas versões antigas por padrão. Se tiver necessidade de verificar essas versões antigas, a solução alternativa é alterar os mapeamentos para definir obsoleto como “falso” ou remover o atributo inteiramente.

Nomenclatura da lista negra e da lista branca:
Seguindo as tendências da área, a equipe de Pesquisa de Segurança de Software, como parte integrante do Micro Focus Fortify, decidiu mudar os termos “lista negra” e “lista branca” para termos mais neutros e com significado específico. Em vez disso, decidimos usar os termos mais descritivos “lista de bloqueios” e “lista de permissões”, respectivamente.
Essa mudança também significa que os seguintes nomes de categoria foram alterados, resultando potencialmente em problemas removidos e novos:

- “Deserialization Bad Practice: Blacklist” agora é “Deserialization Bad Practice: Deny List”
- “Django Bad Practices: Blacklisted Attributes” agora é "Django Bad Practices: Attributes in Deny List"
- “Setting Manipulation: User-Controlled Whitelist” agora é "Setting Manipulation: User-Controlled Allow List"

Criptografia fraca: Atualização de modo inseguro de operação
Acreditamos que não seja mais seguro usar o modo Cipher Block Chaining (CBC) de criptografia simétrica. A partir desta versão, qualquer uso de CBC será relatado como criptografia fraca: Modo inseguro de operação para as bibliotecas com suporte atualmente. Considerando os ataques BEAST, POODLE, Zombie POODLE, GOLDENDOODLE, LUCKY13 e padding oracle, determinamos que o CBC deixou de ser seguro com base em pesquisas criptográficas atualmente conhecidas.

Melhorias de falsos positivos:
Continuamos a ouvir nossos clientes e nos esforçamos para melhorar as taxas de falsos positivos. Durante este lançamento, trabalhamos no seguinte para reduzir o número de falsos positivos:

- Cross-Site Request Forgery
  Redução para contabilizar outros tipos de tokens CSRF e dentro de campos ocultos do ASP.NET
- Server-Side Request Forgery
  Remoções em alguns aplicativos Spring
- Prática deficiente de registro em log: Uso de um fluxo de saída do sistema
  Removido quando o contexto é baseado em aplicativos ou utilitários de linha de comando.
- Código morto: Campo não utilizado
  Dentro de lambdas
- Privacy Violation
  Remoção de problemas duplicados envolvendo senhas em lambdas Java, Scala e Kotlin
- As propriedades do sistema Java agora são tratadas separadamente para propriedades integradas, o que reduz muitos casos de falsos positivos e oferece a possibilidade de encontrar novos vazamentos de informações do sistema.

Visualização:[7]
As regras para referência nula e verificação de nulo redundante foram remodeladas para permitir a redução das taxas de falsos positivos. Em particular, foi adicionada a capacidade de escrever regras personalizadas para lidar com funções internas de verificação de nulo.
No entanto, não está claro se os benefícios são universais por natureza. Portanto, essas melhorias disponíveis no SCA 20.2 ficam desativadas por padrão. Se quiser testar essas melhorias, entre em contato com o Suporte ao Cliente.

[7] Requer SCA 20.2 ou posterior.

Micro Focus Fortify SecureBase [Fortify WebInspect]

O Fortify SecureBase combina verificações de milhares de vulnerabilidades com políticas que orientam os usuários nas seguintes atualizações disponíveis imediatamente pelo SmartUpdate:

Suporte a vulnerabilidades

Weak Cryptographic Signature: User-Controlled Key Source

Ataques de confusão de chave ameaçam a integridade dos JSON Web Tokens (JWT). As implementações dos JWT que não selecionam com cuidado uma fonte de chave para verificar um token recebido do cliente podem permitir que um invasor falsifique o payload do token, ficando vulneráveis a esses ataques. Esta versão inclui uma verificação para avaliar o uso do JWT contra ataques de confusão de chaves.

JSON Web Token: Missing Protection Claims

JSON Web Tokens que não implementam declarações de proteção como “jti”, “exp” e “aud” podem ser suscetíveis a ataques de usurpação de identidade e de reprodução. Esta versão inclui várias verificações que detectam se os tokens encontrados durante uma varredura não possuem essas declarações.

Access control: Authorization Bypass

O JSON Web Token (JWT) fornece proteção contra adulteração de dados porque as informações que ele contém são assinadas digitalmente com o algoritmo HMAC ou RSA. Aplicativos que não conseguem verificar a assinatura podem tornar-se suscetíveis a ataques de acesso não autorizado e de usurpação de identidade. Esta versão inclui várias verificações para avaliar a implementação do JWT para ataques de assinaturas ausentes e arbitrárias.

OAuth2: Insufficient state Parameter Entropy

Implementações do protocolo OAuth2 que não usam um valor de parâmetro de “estado” com entropia suficiente são suscetíveis a ataques de usurpação de identidade. Esta versão inclui uma verificação para avaliar se o parâmetro de estado falha em atender aos requisitos de entropia suficientes.

OAuth2: Insufficient Refresh Token Revocation

Servidores de autorização que não verificam o ID do cliente, o segredo do cliente e a expiração do token de atualização antes de emitir novamente um novo token de acesso podem estar vulneráveis a ataques de usurpação de identidade. Esta versão inclui uma verificação para detectar se um token de atualização pode ser usado para gerar vários tokens de acesso sem o ID e o segredo do cliente.

OAuth2: Insufficient Authorization Code Expiration

Servidores de autorização OAuth2 que não expiram os códigos de autorização no momento certo podem ser suscetíveis a ataques de usurpação de identidade, pois podem continuar concedendo um novo token de acesso válido para códigos de autorização roubados. Esta versão inclui uma verificação para detectar abuso do código de autorização.

Cross-Site Request Forgery

Aplicativos que falham ao usar o parâmetro “estado” no fluxo de autorização do OAuth2 estão suscetíveis a ações não autorizadas de usurpação de identidade de clientes. Além disso, aplicativos que usam armazenamento de cookies para JSON Web Tokens (JWT) correm o risco de sofrer ataques CSRF semelhantes. Esta versão inclui várias verificações que detectam instâncias JWT e OAuth2 vulneráveis a CSRF.

Weak Cryptographic Signature: Insufficient Key Size

JSON Web Tokens contêm informações assinadas digitalmente para comprovar a integridade do payload. Algoritmos simétricos, como HS256 com um segredo fraco, podem sofrer um ataque de força bruta, permitindo que um invasor gere um payload arbitrário e o assine com uma chave válida. Esta versão inclui uma verificação para detectar se um algoritmo fraco como o HS256 é usado para assinar os tokens.

SQL Injection[8]

SQLite é uma biblioteca C que fornece um banco de dados leve baseado em disco que não requer um processo de servidor separado e permite acessar o banco de dados usando uma variante não padrão da linguagem de consulta SQL. Esta versão inclui um aprimoramento da verificação do WebInspect SQL Injection para detectar vulnerabilidades de injeção de SQL em aplicativos da web usando a biblioteca SQLite versão 3.16.0 ou posterior.

Relatório de conformidade

DISA STIG 4.11

Para apoiar nossos clientes governamentais na área de conformidade, este lançamento contém uma correlação das verificações do WebInspect com a versão mais recente da Segurança e Desenvolvimento de Aplicativos da Agência de Sistemas de Informação de Defesa (DISA STIG 4.11).

[8] A inclusão de informações de tipo de banco de dados para SQLite como parte do conteúdo do relatório requer o WebInspect 20.2 ou posterior.

Atualizações da política

DISA STIG 4.11

Foi adicionada uma política personalizada para incluir verificações relevantes para DISA STIG 4.11 à lista existente de políticas com suporte pelo WebInspect SecureBase.

Web API

Foi adicionada uma política personalizada para incluir verificações relevantes para avaliar a segurança de APIs à lista existente de políticas com suporte pelo WebInspect SecureBase.

Erratas diversas:

Nesta versão, continuamos a investir recursos para garantir que possamos reduzir o número de problemas de falsos positivos e melhorar a capacidade dos clientes de auditar problemas. Os clientes também podem esperar alterações nos problemas relatados relacionadas ao seguinte:

Insecure Transport: Weak SSL Cipher:
A verificação identificada pela ID 11285 foi modificada para relatar todas as instâncias de codificação de modo CBC como fracas em comparação com o conteúdo do relatório no Protocolo SSL fraco quando SSLv3 ou TLS1.0 foram detectados junto com codificações CBC. Considerando vulnerabilidades, como POODLE, GoldenPOODLE e ZombieDOODLE, as codificações do modo CBC deixaram de ser recomendadas para a segurança da camada de transporte.

Nomenclatura da lista negra e da lista branca:
Seguindo as tendências da área, a equipe de Pesquisa de Segurança de Software, como parte integrante do Micro Focus Fortify, decidiu mudar os termos “lista negra” e “lista branca” para termos mais neutros e com significado específico. Em vez disso, decidimos usar os termos mais descritivos “lista de bloqueios” (“deny list” substituindo “blacklist”) e “lista de permissões” (“allow list” substituindo “whitelist”), respectivamente, nos conteúdos de verificação de segurança.

Micro Focus Fortify Premium Content

A equipe de pesquisa cria, estende e mantém uma variedade de recursos fora dos nossos principais produtos de inteligência de segurança.

DISA STIG 4.11

Para acompanhar as novas correlações, esta versão também contém um novo pacote de relatórios para o Fortify Software Security Center com suporte para DISA STIG 4.11, que está disponível para download no Portal de Suporte ao Cliente Fortify em Conteúdo Premium.

Micro Focus Fortify Taxonomy: Software Security Errors

O site do Fortify Taxonomy, que contém descrições para suporte de categoria recém-adicionadas, está disponível em https://vulncat.fortify.com/pt-br/. Os clientes que procuram o site anterior com a última atualização com suporte, podem acessá-lo no Portal de suporte do Micro Focus Fortify.

Entre em contato com o suporte técnico do Fortify
Micro Focus Fortify
https://softwaresupport.softwaregrp.com/
+1 (844) 260-7219

SSR de Contato
Alexander M. Hoole
Gerente de Software Security Research
Micro Focus Fortify
hoole@microfocus.com
+1 (650) 258-5916

© Copyright 2020 Micro Focus ou uma de suas afiliadas. As informações contidas neste documento estão sujeitas a alterações sem aviso prévio. As únicas garantias para produtos e serviços da Micro Focus estão estabelecidas nas declarações de garantia expressa que acompanham esses produtos e serviços. Nada neste documento deve ser interpretado como constituindo uma garantia adicional. Micro Focus shall not be liable for technical or editorial errors or omissions contained herein.

25 de setembro de 2020