OWASP
The Open Web Application Security Project

Cornucopia

${Common_Title}

${Common_T00010}

${Common_T00020}
Colin Watson

${Common_T00030}
Colin Watson and Darío De Filippis

${Common_T00040}
Tom Brennan, Johanna Curiel and Timo Goosen

${Common_T00100}

${Common_T00110}

${Common_T00120}

${Common_T00130}

${Common_T00140}

${Common_T00150} ${Common_T00160} ${Common_T00170}

${Common_T00180}

${Common_T00200}

${Common_T00210} ${Common_T00220}

${Common_T00230} ${Common_T00240}

${Common_T00250} ${Common_T00260}

${Common_T00270}

${Common_T00300}

${Common_T00310} ${Common_T00320}

- ${Common_T00330}
- ${Common_T00340}
- ${Common_T00350}
- ${Common_T00360}
- ${Common_T00370}
- ${Common_T00380}

${Common_T00390} ${Common_T00400}

${Common_T00500}

${Common_T00510} ${Common_T00520} ${Common_T00530}

${Common_T00600}

${Common_T00610}

${Common_T00700}

${Common_T00710}

${Common_T00720} ${Common_T00730} ${Common_T00740} ${Common_T00750} ${Common_T00760} ${Common_T00770}

${Common_T00780} ${Common_T00790}

${Common_T00800} ${Common_T00810} ${Common_T00820} ${Common_T00830} ${Common_T00840} ${Common_T00850}

${Common_T00900}

${Common_T00910} ${Common_T00920}

${Common_T01000}

${Common_T01010} ${Common_T01020}

${Common_T01030}

- ${Common_T01040} https://lists.owasp.org/mailman/listinfo/owasp_cornucopia
- ${Common_T01050} https://www.owasp.org/index.php/OWASP_Cornucopia

${Common_T01060} ${Common_T01070}

${Common_T01100}

${Common_T01110} ${Common_T01120} ${Common_T01130} ${Common_T01140} ${Common_T01150}

${Common_T01160}

${Common_T01170} ${Common_T01180}

${Common_T01190}

https://www.owasp.org/index.php/Cornucopia_-_Ecommerce_Website_Edition_-_Wiki_Deck

${Common_T01200}

- ${Common_T01210} https://www.owasp.org/index.php/File:OWASP_SCP_Quick_Reference_Guide_v2.pdf
- ${Common_T01220} https://www.owasp.org/images/3/33/OWASP_Application_Security_Verification_Standard_3.0.1.pdf
- ${Common_T01230} https://www.owasp.org/index.php/AppSensor_DetectionPoints
- ${Common_T01240} http://capec.mitre.org/data/archive/capec_v2.8.zip
- ${Common_T01250} http://www.safecode.org/publications/SAFECode_Agile_Dev_Security0712.pdf

${Common_T01260} ${Common_T01270} ${Common_T01280} ${Common_T01290}

${Common_T01300} ${Common_T01310}

https://youtu.be/i5Y0akWj31khttps://www.owasp.org/index.php/File:Cornucopia-scoresheet.pdf

${Common_T01900}

${Common_T01910} ${Common_T01920} ${Common_T01930}

${Common_T01940}

${Common_T01950} ${Common_T01960} ${Common_T01970}

${Common_T01980}

${Common_T01990} ${Common_T02000}

${Common_T02010} ${Common_T02020} ${Common_T02030}

${Common_T02040}

${Common_T02100}

${Common_T02110} ${Common_T02120}

${Common_T02130} ${Common_T02140}

${Common_T01400}

${Common_T01410}
${Common_T01420}
${Common_T01430}
${Common_T01440}
${Common_T01450}

${Common_T01500}

${Common_T01510} ${Common_T01520} ${Common_T01530}

${Common_T01540}
${Common_T01550}
${Common_T01560}
${Common_T01570}
${Common_T01580} ${Common_T01590}
${Common_T01600}
${Common_T01610}

${Common_T01700}

${Common_T01710}

${Common_T01720}
${Common_T01730}
${Common_T01740}

${Common_T01800}

${Common_T01810}
${Common_T01820}

${Common_T02200}

${Common_T02210}

${Common_T02220}		
${Common_T02230}	${Common_T02270}	${Common_T02310}
${Common_T02240}	*${Common_T02280}*	*${Common_T02320}*
${Common_T02250}	${Common_T02290}	${Common_T02330}
${Common_T02260}	*${Common_T02300}*	*${Common_T02340}*

${Common_T02400}

${Common_T02410}

${Common_T02420}		
${Common_T02430}	${Common_T02470}	${Common_T02510}
${Common_T02440}	*${Common_T02480}*	*${Common_T02520}*
${Common_T02450}	${Common_T02490}	${Common_T02530}
${Common_T02460}	*${Common_T02500}*	*${Common_T02540}*

${Common_T02600}

${Common_T02610}

${Common_T02620}* *${Common_T02630} *${Common_T02640}*

${Common_T02650}

${Common_T02660}

${Common_T02670}

${Common_T02680}* *${Common_T02690} *${Common_T02700}* *${Common_T02710}*
${Common_T02720}* *${Common_T02730} *${Common_T02740}*

${Common_T02750}

${Common_T02760}* *${Common_T02770} *${Common_T02780}*

${Common_T02790}

${Common_T02800}* *${Common_T02810}

${Common_T02820}

${Common_T02830}* *${Common_T02840}

${Common_T02850}

${Common_T02860}* *${Common_T02870} *${Common_T02880}*

${Common_T02890}

${Common_T02900}* *${Common_T02910} *${Common_T02920}*

${Common_T02930}

${Common_T02940}* *${Common_T02950} *${Common_T02960}* *${Common_T02970}*
${Common_T02980}* *${Common_T02990}

${Common_T03000}

${Common_T03010}* *${Common_T03020}

${Common_T03030}

${Common_T03040}* *${Common_T03050}

https://www.owasp.org/index.php/Cornucopia_-_Ecommerce_Website_Edition_-_Wiki_Deck

${Common_T03060}* *${Common_T03070}

https://www.owasp.org/index.php/OWASP_Cornucopia - tab=FAQs

DATA VALIDATION & ENCODING

A

Você inventou um novo ataque contra a Validação de Dados de Entrada e Codificação de Dados de Saída

Leia mais sobre este tópico em OWASP Cheat Sheets. Pesquise sobre validação dos dados de entrada, Prevenção de XSS(Cross-site Scripting), Prevenção do DOM baseado em XSS, Prevenção de SQL Injection e Parametrização de Consultas

DATA VALIDATION & ENCODING

(${Common_NoCard})

DATA VALIDATION & ENCODING

2

Brian consegue reunir o básico de informações sobre a utilização e configuração de base de dados, lógica, codificação, além da utilização de softwares, serviços e infraestrutura nas mensagens de erro ou em mensagens de configuração, ou na presença de arquivos de instalação (padrões ou antigos), ou em evidências de testes, ou em backups ou em exposição de código fonte

OWASP SCP
69, 107-109, 136-137, 153, 156, 158, 162

OWASP ASVS
1.10, 4.5, 8.1, 11.5, 19.1, 19.5

OWASP APPSENSOR
HT1-3

CAPEC
54, 541

SAFECODE
4, 23

DATA VALIDATION & ENCODING

3

Robert consegue inserir dados maliciosos pois o formato de protocolo não foi checado, ou duplicações são aceitas, ou a estrutura não está sendo verificada, ou os dados individuais não foram validados por formato, tipo, intervalo, tamanho e por uma lista de caracteres ou formatos possíveis

OWASP SCP
-

OWASP ASVS
5.1, 5.16, 5.17, 5.18, 5.19, 5.20, 11.1, 11.2

OWASP APPSENSOR
RE7-8, AE4-7, IE2-3, CIE1, CIE3-4, HT1-3

CAPEC
28, 48, 126, 165, 213, 220-221, 261-262, 271-272

SAFECODE
3, 16, 24, 35

DATA VALIDATION & ENCODING

4

Dave consegue inserir nomes ou dados de campos mal intencionados porque isto não está sendo verificado no contexto de cada usuário e processo

OWASP SCP
8, 10, 183

OWASP ASVS
4.16, 5.16, 5.17, 15.1

OWASP APPSENSOR
RE3-6, AE8-11, SE1, SE3-6, IE2-4, HT1-3

CAPEC
28, 31, 48, 126, 162, 165, 213, 220-221, 261

SAFECODE
24, 35

DATA VALIDATION & ENCODING

5

Jee consegue ignorar as rotinas centralizadas de codificação de saída pois elas não estão sendo usadas em todos os lugares, ou a codificação errada está sendo usada

OWASP SCP
3, 15, 18-22, 168

OWASP ASVS
1.7, 5.15, 5.21, 5.22, 5.23

OWASP APPSENSOR
-

CAPEC
28, 31, 152, 160, 468

SAFECODE
2, 17

DATA VALIDATION & ENCODING

6

Jason consegue ignorar as rotinas centralizadas de validação de dados de entrada pois elas não estão sendo usadas em todos os campos de entrada de dados

OWASP SCP
3, 168

OWASP ASVS
1.7, 5.6, 5.19

OWASP APPSENSOR
IE2-3

CAPEC
28

SAFECODE
3, 16, 24

DATA VALIDATION & ENCODING

7

Jan consegue carregar/enviar informações especiais visando evitar validações de campos porque o conjunto de caracteres não é especificado e aplicado, ou o dado de entrada é codificado diversas vezes, ou o dado não é totalmente convertido no mesmo formado que a aplicação usa (ex: canonicalização) antes da validação, ou as variáveis não são fortemente tipadas

OWASP SCP
4-5, 7, 150

OWASP ASVS
5.6, 11.8

OWASP APPSENSOR
IE2-3, EE1-2

CAPEC
28, 153, 165

SAFECODE
3, 16, 24

DATA VALIDATION & ENCODING

8

Sarah consegue ignorar as rotinas centralizadas de tratamento (sanitização) pois elas não estão sendo usadas de forma abrangente

OWASP SCP
15, 169

OWASP ASVS
1.7, 5.21, 5.23

OWASP APPSENSOR
-

CAPEC
28, 31, 152, 160, 468

SAFECODE
2, 17

DATA VALIDATION & ENCODING

9

Shamun consegue ignorar as verificações de validação de entrada ou de saída porque as falhas de validação não são rejeitadas e/ou tratadas (sanitização)

OWASP SCP
6, 21-22, 168

OWASP ASVS
5.3

OWASP APPSENSOR
IE2-3

CAPEC
28

SAFECODE
3, 16, 24

DATA VALIDATION & ENCODING

10

Dario consegue explorar a confiabilidade da aplicação em fonte de dados (ex: dados definidos pelo usuário, manipulação de dados armazenados localmente, mudança do estado dos dados em dispositivos clientes, falta de verificação da identidade durante uma validação de dados, como Dario pode fingir ser Colin)

OWASP SCP
2, 19, 92, 95, 180

OWASP ASVS
5.19, 10.6, 16.2, 16.3, 16.4, 16.5, 16.8

OWASP APPSENSOR
IE4, IE5

CAPEC
12, 51, 57, 90, 111, 145, 194-195, 202, 218, 463

SAFECODE
14

DATA VALIDATION & ENCODING

J

Dennis tem o controle sobre validações de entrada de dados, validações de saída de dados ou codificação de saída ou rotinas que ele consegue ignorar/burlar

OWASP SCP
1, 17

OWASP ASVS
5.5, 5.18

OWASP APPSENSOR
RE3, RE4

CAPEC
87, 207, 554

SAFECODE
2, 17

DATA VALIDATION & ENCODING

Q

Geoff consegue injetar dados num dispositivo ou num interpretador no lado do cliente porque uma interface parametrizada não foi usada, ou não foi implementada corretamente, ou os dados não foram codificados corretamente para o contexto proposto, ou não há uma política restritiva para a codificação ou a inclusão de dados

OWASP SCP
10, 15-16, 19-20

OWASP ASVS
5.15, 5.22, 5.23, 5.24, 5.25

OWASP APPSENSOR
IE1, RP3

CAPEC
28, 31, 152, 160, 468

SAFECODE
2, 17

DATA VALIDATION & ENCODING

K

Gabe consegue injetar dados num interpretador no lado do servidor (ex: SQL, comandos para o sistema operacional, Xpath, Server JavaScript, SMTP) porque uma interface parametrizada não foi usada ou não foi implementada corretamente

OWASP SCP
15, 19-22, 167, 180, 204, 211-212

OWASP ASVS
5.10, 5.11, 5.12, 5.13, 5.14, 5.16, 5.21

OWASP APPSENSOR
CIE1, CIE2

CAPEC
23, 28, 76, 152, 160, 261

SAFECODE
2, 19-20

(${Common_NoCard})

(${Common_NoCard})

AUTHENTICATION

A

Você inventou um novo ataque contra a Autenticação e Gerenciamento de Credenciais

Leia mais sobre este tópico em OWASP Authentication Cheat Sheet

AUTHENTICATION

(${Common_NoCard})

AUTHENTICATION

2

James pode assumir as funções de autenticação sem que o usuário real esteja ciente do uso destas funções (ex: tente fazer login, logar com credenciais, redefinir a senha)

OWASP SCP 47, 52
OWASP ASVS 2.12, 8.4, 8.10
OWASP APPSENSOR UT1
CAPEC -
SAFECODE 28

AUTHENTICATION

3

Muhammad consegue obter a senha de um usuário ou outros dados, pela observação durante a autenticação, ou cache local, ou pela memória, ou pelo tráfego de dados, ou pela leitura de algum local desprotegido, ou porque isto é amplamente conhecido, ou porque não há expiração de dados, ou por que o usuário não consegue trocar sua própria senha

OWASP SCP 36-37, 40, 43, 48, 51, 119, 139-140, 146
OWASP ASVS 2.2, 2.17, 2.24, 8.7, 9.1, 9.4, 9.5, 9.9, 9.11
OWASP APPSENSOR -
CAPEC 37, 546
SAFECODE 28

AUTHENTICATION

4

Sebastien pode identificar facilmente nomes de usuários ou consegue elencar quem eles são

OWASP SCP 33, 53
OWASP ASVS 2.18, 2.28
OWASP APPSENSOR AE1
CAPEC 383
SAFECODE 28

AUTHENTICATION

5

Javier pode usar credenciais padrões (default), de teste ou facilmente adivinhadas para autenticação, ou consegue autenticar através de contas inativas ou autentica-se por contas não necessariamente da aplicação

OWASP SCP 54, 175, 178
OWASP ASVS 2.19
OWASP APPSENSOR AE12, HT3
CAPEC 70
SAFECODE 28

AUTHENTICATION

6

Sven consegue reutilizar uma senha temporária porque o usuário não precisa troca-la no primeiro acesso, ou o tempo de expiração é muito longo, ou o tempo de expiração não existe, ou não é usado um método de entrega out-of-band (ex: aplicação mobile, SMS)

OWASP SCP 37, 45-46, 178
OWASP ASVS 2.22
OWASP APPSENSOR -
CAPEC 50
SAFECODE 28

AUTHENTICATION

7

Cecilia consegue usar força bruta e ataques de dicionário (dictionary attacks) contra uma ou muitas contas sem limitação, ou estes ataques são simplificados pois as senhas tem baixa complexidade, tamanho reduzido, inexistência de expiração e regras para reuso

OWASP SCP 33, 38-39, 41, 50, 53
OWASP ASVS 2.7, 2.20, 2.23, 2.25, 2.27
OWASP APPSENSOR AE2, AE3
CAPEC 2, 16
SAFECODE 27

AUTHENTICATION

8

Kate consegue ignorar a autenticação porque isto não é uma falha de segurança (ex: o acesso sem autenticação está assinalado como padrão)

OWASP SCP 28
OWASP ASVS 2.6
OWASP APPSENSOR -
CAPEC 115
SAFECODE 28

AUTHENTICATION

9

Claudia consegue assumir funções críticas porque os requisitos de autenticação são muito fracos (ex: não é usado autenticação com força de senha), ou não é um requisito revalidar a autenticação com frequência

OWASP SCP 55-56
OWASP ASVS 2.1, 2.9, 2.26, 2.31, 4.15
OWASP APPSENSOR -
CAPEC 21
SAFECODE 14, 28

AUTHENTICATION

10

Pravin consegue ignorar controle de autenticação porque não está sendo usado um módulo/framework/serviço de autenticação que seja centralizado, testado, comprovado e aprovado para gerir requisições

OWASP SCP 25-27
OWASP ASVS 1.7, 2.30
OWASP APPSENSOR -
CAPEC 90, 115
SAFECODE 14, 28

AUTHENTICATION

J

Mark consegue acessar recursos ou serviços porque não há requisitos de autenticação, ou, por engano, um outro sistema ou outra ação realizou autenticação

OWASP SCP 23, 32, 34
OWASP ASVS 2.1
OWASP APPSENSOR -
CAPEC 115
SAFECODE 14, 28

AUTHENTICATION

Q

Jaime consegue ignorar a autenticação porque não é aplicado o mesmo rigor para todas as funções de autenticação (ex: logar, troca de senha, recuperação de senha, logout, acesso administrador) ou não é aplicado o mesmo rigor nos diversos locais de acesso e versões do sistema(ex:mobile website, mobile app, full website, API, call center)

OWASP SCP 23, 29, 42, 49
OWASP ASVS 2.1, 2.8
OWASP APPSENSOR -
CAPEC 36, 50, 115, 121, 179
SAFECODE 14, 28

AUTHENTICATION

K

Olga consegue influenciar ou alterar o código ou a rotina de autenticação e com isto ignorar a autenticação

OWASP SCP 24
OWASP ASVS 2.4, 13.2
OWASP APPSENSOR -
CAPEC 115, 207, 554
SAFECODE 14, 28

(${Common_NoCard})

(${Common_NoCard})

SESSION MANAGEMENT

A

Você inventou um novo ataque contra o Gerenciamento de Sessões

Leia mais sobre este tópico em OWASP Session Management Cheat Sheet e prevenção de ataques do tipo Cross Site Request Forgery (CSRF)

SESSION MANAGEMENT

(${Common_NoCard})

SESSION MANAGEMENT

2

William tem o controle sobre a geração de identificadores de sessão

OWASP SCP 58-59
OWASP ASVS 3.10
OWASP APPSENSOR SE2
CAPEC 31, 60-61
SAFECODE 28

SESSION MANAGEMENT

3

Ryan consegue usar uma única conta em paralelo, pois as sessões simultâneas são permitidas

OWASP SCP 68
OWASP ASVS 3.16, 3.17, 3.18
OWASP APPSENSOR -
CAPEC -
SAFECODE 28

SESSION MANAGEMENT

4

Alison consegue configurar identificadores de cookies em outras aplicações web porque o domínio ou o caminho não são suficientemente limitados

OWASP SCP 59, 61
OWASP ASVS 3.12
OWASP APPSENSOR SE2
CAPEC 31, 61
SAFECODE 28

SESSION MANAGEMENT

5

John consegue prever ou adivinhar identificadores de sessão porque estes não são alterados quando uma regra de usuário é alterada (ex: antes e depois da autenticação) e quando uma troca entre meios de comunicação criptografados e não criptografados acontece, ou os identificadores são curtos e não randômicos, ou não são modificados periodicamente

OWASP SCP 60, 62, 66-67, 71-72
OWASP ASVS 3.2, 3.7, 3.11
OWASP APPSENSOR SE4-6
CAPEC 31
SAFECODE 28

SESSION MANAGEMENT

6

Gary consegue ter o controle da sessão de um usuário porque o tempo de encerramento(timeout) da sessão é longo ou inexiste, ou o tempo limite da sessão é longo ou inexiste, ou a mesma sessão pode ser usada para mais de um dispositivo/local

OWASP SCP 64-65
OWASP ASVS 3.3, 3.4, 3.16, 3.17, 3.18
OWASP APPSENSOR SE5, SE6
CAPEC 21
SAFECODE 28

SESSION MANAGEMENT

7

Casey consegue utilizar a sessão de Adam depois dele ter finalizado o uso da aplicação, porque a função de logout inexiste, ou Adam não fez logout, ou a função de logout não termina a sessão de forma adequada

OWASP SCP 62-63
OWASP ASVS 3.2, 3.5
OWASP APPSENSOR -
CAPEC 21
SAFECODE 28

SESSION MANAGEMENT

8

Matt consegue utilizar longas sessões porque a aplicação não solicita uma nova autenticação de forma periódica para validar se os privilégios do usuário foram alterados

OWASP SCP 96
OWASP ASVS -
OWASP APPSENSOR -
CAPEC 21
SAFECODE 28

SESSION MANAGEMENT

9

Ivan consegue roubar identificadores de sessão porque estes são transmitidos em canais inseguros, ou estão logados, ou são exibidos em mensagens de erros, ou estão em URLs, ou são acessíveis pelo código que o atacante consegue alterar ou influenciar

OWASP SCP 69, 75-76, 119, 138
OWASP ASVS 3.6, 8.7, 10.3
OWASP APPSENSOR SE4-6
CAPEC 31, 60
SAFECODE 28

SESSION MANAGEMENT

10

Marce consegue inventar requisições porque tokens randômicos e fortes (ou seja, tokens anti-CSRF) ou similares não estão sendo usados para ações que mudam estado. Estas requisições podem ser por sessão ou por requisição (request) em ações mais críticas

OWASP SCP 73-74
OWASP ASVS 4.13
OWASP APPSENSOR IE4
CAPEC 62, 111
SAFECODE 18

SESSION MANAGEMENT

J

Jeff consegue reenviar uma interação de repetição idêntica (ex: requisição HTTP, sinal, botão pressionado) e ela é aceita, sem rejeição

OWASP SCP -
OWASP ASVS 15.1, 15.2
OWASP APPSENSOR IE5
CAPEC 60
SAFECODE 12, 14

SESSION MANAGEMENT

Q

Salim consegue ignorar o gerenciamento de sessão porque este não é aplicado de forma abrangente e consistente por toda a aplicação

OWASP SCP 58
OWASP ASVS 3.1
OWASP APPSENSOR -
CAPEC 21
SAFECODE 14, 28

SESSION MANAGEMENT

K

Peter consegue ignorar o controle de gerenciamento de sessão porque este foi autoconstruído e/ou é fraco, ao invés de ter sido usado a estrutura padrão de um framework ou um modulo testado e aprovado

OWASP SCP 58, 60
OWASP ASVS 1.7
OWASP APPSENSOR -
CAPEC 21
SAFECODE 14, 28

(${Common_NoCard})

(${Common_NoCard})

AUTHORIZATION

A

Você inventou um novo ataque contra Controle de Acessos

Leia mais sobre este tópico em OWASP Development Guide e OWAPS Testing Guide

AUTHORIZATION

(${Common_NoCard})

AUTHORIZATION

2

Tim consegue alterar nomes/endereços (paths) onde os dados são enviados ou encaminhados para alguém

OWASP SCP 44
OWASP ASVS 4.1, 4.16, 16.1
OWASP APPSENSOR -
CAPEC 153
SAFECODE 8, 10-11

AUTHORIZATION

3

Christian consegue acessar informações, que ele não deveria ter permissão, por meio de outro mecanismo que tenha permissão (ex: indexador de pesquisa, log, relatórios) ou porque a informação está armazenada em cache, ou mantida por mais tempo do que o necessário, ou outra vazamento de informação

OWASP SCP 51, 100, 135, 139-141, 150
OWASP ASVS 4.1, 8.2, 9.1-9.6, 9.11, 16.6-16.7
OWASP APPSENSOR -
CAPEC 69, 213
SAFECODE 8, 10-11

AUTHORIZATION

4

Kelly consegue ignorar controles de acesso porque estes não falham seguramente (ex: a permissão de acesso está assinalada como padrão)

OWASP SCP 79-80
OWASP ASVS 4.8
OWASP APPSENSOR -
CAPEC 122
SAFECODE 8, 10-11

AUTHORIZATION

5

Chad consegue acessar recursos que não deveria ter acesso devido a inexistência de uma autorização ou por concessão de privilégios excessivos (ex: não usar o princípio de menor privilégio possível). Os recursos podem ser serviços, processos, AJAX, Flash, vídeo, imagens, documentos, arquivos temporários, dados de sessão, propriedades do sistema, dados de configuração, logs

OWASP SCP 70, 81, 83-4, 87-9, 99, 117, 131-2, 142, 154, 170, 179
OWASP ASVS 4.1, 4.4, 4.9, 19.3
OWASP APPSENSOR ACE1, ACE2, ACE3, ACE4, HT2
CAPEC 75, 87, 95, 126, 149, 155, 203, 213, 264-265
SAFECODE 8, 10-11, 13

AUTHORIZATION

6

Eduardo consegue acessar dados que ele não tem permissão embora ele tem permissão em formulários, páginas, URL ou pontos de entrada

OWASP SCP 81, 88, 131
OWASP ASVS 4.1, 4.4
OWASP APPSENSOR ACE1-4
CAPEC 122
SAFECODE 8, 10-11

AUTHORIZATION

7

Yuanjing consegue acessar funções, telas e propriedades do aplicativo, a qual ele não está autorizado a ter acesso

OWASP SCP 81, 85-86, 131
OWASP ASVS 4.1, 4.4
OWASP APPSENSOR ACE1-4
CAPEC 122
SAFECODE 8, 10-11

AUTHORIZATION

8

Tom consegue ignorar regras de negócios alterando o fluxo/sequência usual do processo, ou realizando o processo na forma incorreta, ou manipulando valores de data e hora usados pela aplicação, ou usando recursos válidos para fins não intencionais, ou pela manipulação incorreta do controle de dados

OWASP SCP 10, 32, 93-94, 189
OWASP ASVS 4.10, 4.15, 4.16, 8.13, 15.1
OWASP APPSENSOR ACE3
CAPEC 25, 39, 74, 162, 166, 207
SAFECODE 8, 10-12

AUTHORIZATION

9

Mike consegue usar indevidamente uma aplicação quando uma funcionalidade é usada de forma muito rápida, ou com muita frequência, ou de outra maneira a qual a funcionalidade não se destina, ou pelo consumo de recursos da aplicação ou pela condição de corrida (race conditions) ou utilização excessiva da funcionalidade

OWASP SCP 94
OWASP ASVS 4.14, 15.2
OWASP APPSENSOR AE3, FIO1-2, UT2-4, STE1-3
CAPEC 26, 29, 119, 261
SAFECODE 1, 35

AUTHORIZATION

10

Richard consegue ignorar os controles de acesso centralizados pois estes não estão sendo utilizados de forma abrangente em todas as interações

OWASP SCP 78, 91
OWASP ASVS 1.7, 4.11
OWASP APPSENSOR ACE1-4
CAPEC 36, 95, 121, 179
SAFECODE 8, 10-11

AUTHORIZATION

J

Dinis consegue acessar informações referente a configurações de segurança ou consegue acessar a lista de controle de acesso

OWASP SCP 89-90
OWASP ASVS 4.10, 13.2
OWASP APPSENSOR -
CAPEC 75, 133, 203
SAFECODE 8, 10-11

AUTHORIZATION

Q

Christopher consegue injetar um comando que a aplicação vai executar no mais alto nível de privilégio

OWASP SCP 209
OWASP ASVS 5.12
OWASP APPSENSOR -
CAPEC 17, 30, 69, 234
SAFECODE 8, 10-11

AUTHORIZATION

K

Ryan consegue influenciar ou alterar controles de acesso e permissões e consegue ignora-los

OWASP SCP 77, 89, 91
OWASP ASVS 4.9, 4.10, 13.2
OWASP APPSENSOR -
CAPEC 207, 554
SAFECODE 8, 10-11

(${Common_NoCard})

(${Common_NoCard})

CRYPTOGRAPHY

A

Você inventou um novo ataque contra Práticas de Criptografia

Leia mais sobre este tópico em OWASP Cryptographic Storage Cheat Sheet e OWASP Transport Layer Protection Cheat Sheet

OWASP SCP
105, 133, 135

OWASP ASVS
-

OWASP APPSENSOR
-

CAPEC
-

SAFECODE
21, 29

CRYPTOGRAPHY

(${Common_NoCard})

CRYPTOGRAPHY

2

Kyun consegue acesso a dados porque isto foi ocultado/ofuscado/escondido ao invés de ser usada uma função de criptografia aprovada

OWASP SCP
92, 205, 212

OWASP ASVS
8.11, 11.7, 13.2, 19.5, 19.6, 19.7, 19.8

OWASP APPSENSOR
SE1, IE4

CAPEC
31, 39, 68, 75, 133, 145, 162, 203, 438-439, 442

SAFECODE
12, 14

CRYPTOGRAPHY

3

Axel consegue modificar dados que estão armazenados ou que são temporários ou transitórios, ou consegue modificar código fonte, ou consegue modificar patches/atualizações, ou alterar dados de configuração, pois a integridade não foi checada

CRYPTOGRAPHY

4

Paulo consegue acesso a dados transitórios não criptografados, embora o canal de comunicação esteja criptografado

OWASP SCP
37, 88, 143, 214

OWASP ASVS
7.12, 9.2

OWASP APPSENSOR
-

CAPEC
185-187

SAFECODE
14, 29-30

CRYPTOGRAPHY

5

Kyle consegue ignorar controles criptográficos porque eles não falham de forma segura (ex: eles são desprotegidos por padrão)

OWASP SCP
103, 145

OWASP ASVS
7.2, 10.3

OWASP APPSENSOR
-

CAPEC
-

SAFECODE
21, 29

CRYPTOGRAPHY

6

Romain consegue ler e modificar dados descriptografados que estão na memória ou são transitórios (ex: credenciais, identificadores de sessão, dados pessoais e comercialmente relevantes), em uso ou em comunicação dentro da aplicação, ou entre aplicação e usuário, ou entre a aplicação e sistemas externos

OWASP SCP
36-37, 143, 146-147

OWASP ASVS
2.16, 9.2, 9.11, 10.3, 19.2

OWASP APPSENSOR
-

CAPEC
31, 57, 102, 157-158, 384, 466, 546

SAFECODE
29

CRYPTOGRAPHY

7

Gunter consegue interceptar ou modificar dados criptografados em trânsito porque o protocolo está mal implantado, ou configurado de forma fraca, ou os certificados estão inválidos, ou os certificados não são confiáveis, ou a conexão pode ser deteriorada para uma comunicação mais fraca ou descriptografada

OWASP SCP
75, 144-145, 148

OWASP ASVS
10.1, 10.5, 10.10, 10.11, 10.12, 10.13, 10.14

OWASP APPSENSOR
IE4

CAPEC
31, 216

SAFECODE
14, 29-30

CRYPTOGRAPHY

8

Eoin consegue acesso a dados de negócios armazenados (ex: senhas, identificadores de sessão, informações de identificação pessoal - PII, dados de titular de cartão) pois estes dados não estão criptografados de forma segura ou com segurança

OWASP SCP 30-31, 70, 133, 135
OWASP ASVS 2.13, 7.7, 7.8, 9.2
OWASP APPSENSOR -
CAPEC 31, 37, 55
SAFECODE 21, 29, 31

CRYPTOGRAPHY

9

Andy consegue ignorar a geração de números aleatórios/randômicos, ou ignorar a geração aleatória de GUID, ou ignorar as funções de criptografia e hashing porque eles são fracos ou foram autoconstruídos

OWASP SCP 60, 104-105
OWASP ASVS 7.6, 7.7, 7.8, 7.15
OWASP APPSENSOR -
CAPEC 97
SAFECODE 14, 21, 29, 32-33

CRYPTOGRAPHY

10

Susanna consegue quebrar a criptografia em uso pois a criptografia não é forte o suficiente para oferecer a proteção exigida, ou esta não é forte o suficiente para tratar a quantidade de esforço que o atacante está disposto a fazer

OWASP SCP 104-105
OWASP ASVS -
OWASP APPSENSOR -
CAPEC 97, 463
SAFECODE 14, 21, 29, 31-33

CRYPTOGRAPHY

J

Justin consegue ler credenciais para acessar recursos internos e externos, serviços e outros sistemas porque estas credenciais estão armazenadas num formato descriptografado ou salvos no código fonte

OWASP SCP 35, 90, 171-172
OWASP ASVS 2.29
OWASP APPSENSOR -
CAPEC 116
SAFECODE 21, 29

CRYPTOGRAPHY

Q

Randolph consegue acessar ou prever os dados mestres de criptografia

OWASP SCP 35, 102
OWASP ASVS 7.8, 7.9, 7.11, 7.13, 7.14
OWASP APPSENSOR -
CAPEC 116-117
SAFECODE 21, 29

CRYPTOGRAPHY

K

Dan consegue influenciar ou alternar as rotinas/codificações de criptografia (encriptação, hashing, assinaturas digitais, números aleatórios e geração de GUID) e consegue ignorá-los também

OWASP SCP 31, 101
OWASP ASVS 7.11
OWASP APPSENSOR -
CAPEC 207, 554
SAFECODE 14, 21, 29

(${Common_NoCard})

(${Common_NoCard})

CORNUCOPIA

A

Você inventou um novo ataque de qualquer tipo

Leia mais sobre segurança da aplicação nos guias da OWASP (Requirements, Development, Code Review and Testing) e na série OWASP Cheat Sheet, e no modelo de maturidade Open SAMM (Software Assurance Maturity Model)

CORNUCOPIA

(${Common_NoCard})

CORNUCOPIA

2

Lee consegue ignorar os controles do aplicativo pois foram usadas funções arriscadas da linguagem de programação ao invés de opções seguras, ou há erros de conversão, ou porque o aplicativo está inseguro quando um recurso externo está indisponível, ou há race condition, ou há problemas na inicialização ou alocação de recursos, ou quando há sobrecarga

OWASP SCP
194-202, 205-209

OWASP ASVS
5.1

OWASP APPSENSOR
-

CAPEC
25-26, 29, 96, 123-124, 128-129, 264-265

SAFECODE
3, 5-7, 9, 22, 25-26, 34

CORNUCOPIA

3

Andrew consegue acessar o código fonte, ou descompilar o aplicativo, ou consegue acessar a lógica do negócio para entender como a aplicação funciona e quais segredos ela contêm

OWASP SCP
134

OWASP ASVS
19.5

OWASP APPSENSOR
-

CAPEC
189, 207

SAFECODE
-

CORNUCOPIA

4

Keith consegue realizar uma ação e isto não é atribuído a ele

OWASP SCP
23, 32, 34, 42, 51, 181

OWASP ASVS
8.10

OWASP APPSENSOR
-

CAPEC
-

SAFECODE
-

CORNUCOPIA

5

Larry consegue induzir a confiança de outras partes, incluindo usuários autenticados, ou violar esta confiança em outro lugar (ex: em outro aplicativo)

OWASP SCP
-

OWASP ASVS
-

OWASP APPSENSOR
-

CAPEC
89, 103, 181, 459

SAFECODE
-

CORNUCOPIA

6

Aaron consegue ignorar os controles porque a manipulação de erros/exceções é perdida/ignorada, ou é implementada de forma inconsistente ou parcial, ou não há negação de acesso por padrão (ex: erros devem terminar o acesso/execução da funcionalidade), ou depende do tratamento por algum outro serviço ou sistema

OWASP SCP
109-112, 155

OWASP ASVS
8.2, 8.4

OWASP APPSENSOR
-

CAPEC
54, 98, 164

SAFECODE
4, 11, 23

CORNUCOPIA

7

As ações de Mwengu não podem ser investigadas porque não há um registro correto de eventos de segurança com precisão, ou não há uma trilha de auditoria completa, ou estas podem ser alteradas ou excluídas pelo Mwengu, ou não existe um serviço de registro centralizado

OWASP SCP
113-115, 117-118, 121-130

OWASP ASVS
2.12, 8.3, 8.4, 8.5, 8.6, 8.7, 8.8, 8.9, 8.10, 8.11, 8.12, 9.10, 10.4

OWASP APPSENSOR
-

CAPEC
93

SAFECODE
4

CORNUCOPIA

8

David consegue ignorar o aplicativo para obter acesso aos dados porque a infraestrutura de rede e servidores e os serviços suportados não foram configurados de forma segura, as configurações não são verificadas periodicamente e os patches de segurança não são aplicados, ou os dados armazenados localmente não são fisicamente protegidos

OWASP SCP
151-152, 156, 160-161, 173-177

OWASP ASVS
19.1, 19.4, 19.6, 19.7, 19.8

OWASP APPSENSOR
RE1, RE2

CAPEC
37, 220, 310, 436, 536

SAFECODE
-

CORNUCOPIA

9

Michael consegue ignorar o aplicativo para obter acesso aos dados porque ferramentas ou interfaces administrativas não estão adequadamente seguras

OWASP SCP
23, 29, 56, 81-82, 84-90

OWASP ASVS
2.1, 2.32

OWASP APPSENSOR
-

CAPEC
122, 233

SAFECODE
-

CORNUCOPIA

10

Xavier consegue contornar os controles do aplicativo porque os códigos fontes tanto dos frameworks, como de bibliotecas e componentes utilizados contêm código malicioso ou vulnerabilidades

OWASP SCP
57, 151-152, 204-205, 213-214

OWASP ASVS
1.11

OWASP APPSENSOR
-

CAPEC
68, 438-439, 442, 524, 538

SAFECODE
15

CORNUCOPIA

J

Roman consegue explorar o aplicativo pois este foi compilado usando ferramentas desatualizadas ou configurações não seguras como padrão ou informações de segurança não foram documentadas e passadas para o time operacional

OWASP SCP
90, 137, 148, 151-154, 175-179, 186, 192

OWASP ASVS
19.5, 19.9

OWASP APPSENSOR
-

CAPEC
-

SAFECODE
4

CORNUCOPIA

Q

Jim pode realizar ações mal-intencionadas, não normais, sem detecção e resposta em tempo real pela aplicação

OWASP SCP
-

OWASP ASVS
4.14, 9.8, 15.1, 15.2

OWASP APPSENSOR
(All)

CAPEC
-

SAFECODE
1, 27

CORNUCOPIA

K

Gareth pode utilizar o aplicativo para negar o serviço a alguns ou a todos os usuários

OWASP SCP
41, 55

OWASP ASVS
-

OWASP APPSENSOR
UT1-4, STE3

CAPEC
2, 25, 119, 125

SAFECODE
1

JOKER

Joker

Alice consegue utilizar a aplicação para realizar ataques a dados e usuários do sistema

Você pensou em se tornar membro individual da OWASP? Todas as ferramentas, guias e reuniões locais são gratuitas para todos, mas ser um membro individual apoia o trabalho da OWASP

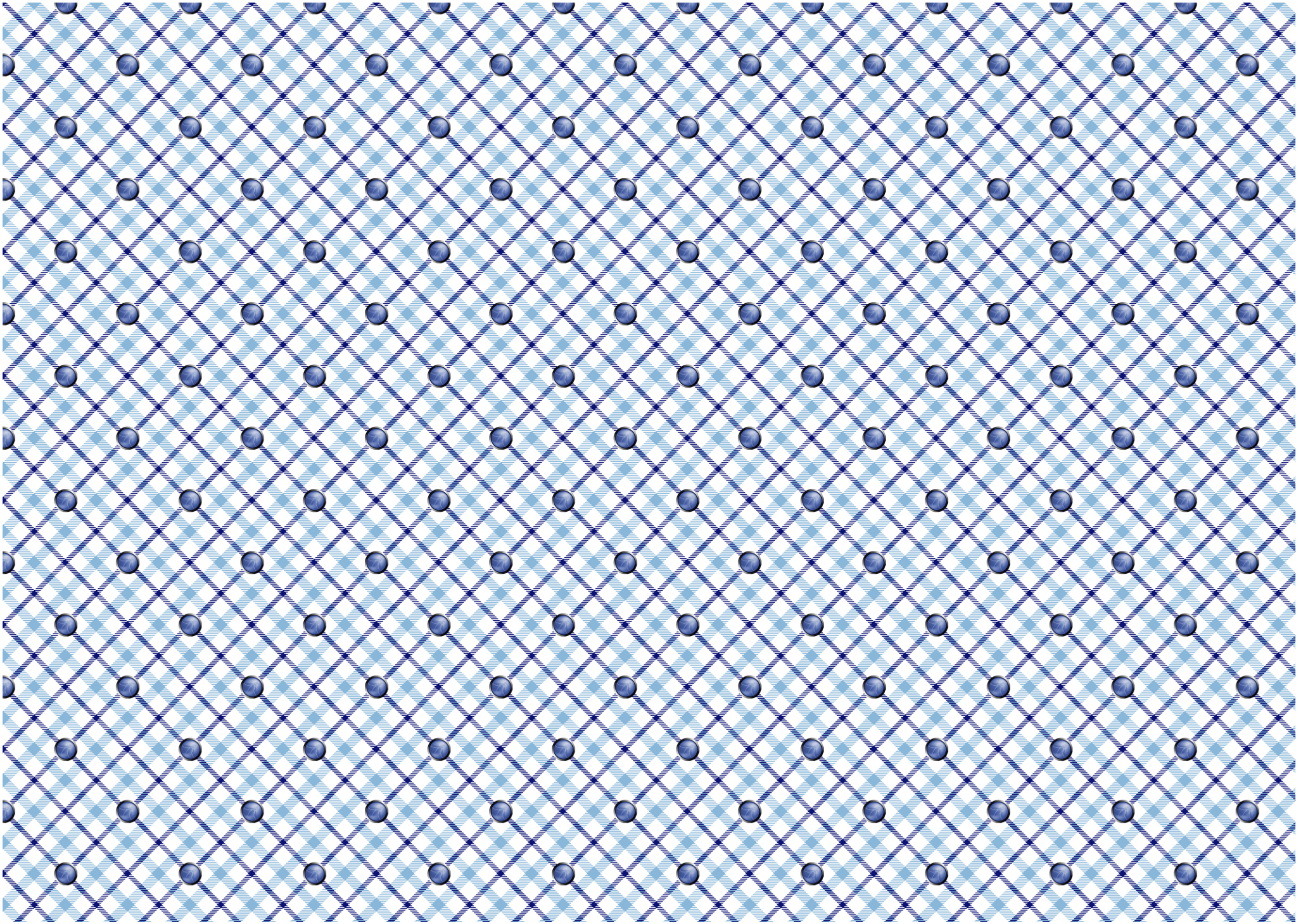
JOKER

Joker

Bob pode influenciar, alterar ou mudar a aplicação para que ela não cumpra os propósitos legais, regulamentadores, contratuais ou outras diretrizes organizacionais

Examine as vulnerabilidades e descubra como elas podem ser solucionadas através do aplicativo de treinamento OWASP Broken Web Applications VM, ou usando o desafio online Hacking Lab. Ambos são gratuitos

Cut here



${Common_T03100}

${Common_T03110}		${Common_T03120}
${Common_T03130}	30 Jul 2012	${Common_T03140}
${Common_T03150}	10 Aug 2012	${Common_T03160}
${Common_T03170}	15 Aug 2012	${Common_T03180}
${Common_T03190}	25 Feb 2013	${Common_T03200} ${Common_T03210} ${Common_T03220} ${Common_T03230}
${Common_T03240}	25 Feb 2013	${Common_T03250}
${Common_T03260}	03 Jun 2013	${Common_T03270} ${Common_T03280} ${Common_T03290} ${Common_T03300} ${Common_T03310} ${Common_T03320} ${Common_T03330} ${Common_T03340}
${Common_T03350}	14 Aug 2013	${Common_T03360} ${Common_T03370} ${Common_T03380} ${Common_T03390} ${Common_T03400} ${Common_T03410}
${Common_T03420}	18 Sep 2013	${Common_T03430} ${Common_T03440} ${Common_T03450} ${Common_T03460}
${Common_T03470}	01 Feb 2014	${Common_T03480}
${Common_T03490}	21 Mar 2014	${Common_T03500} ${Common_T03510} ${Common_T03520} ${Common_T03530}
${Common_T03540}	04 Mar 2015	${Common_T03550} ${Common_T03560} ${Common_T03570}
${Common_T03580}	29 Jun 2016	${Common_T03590} ${Common_T03600} ${Common_T03610} ${Common_T03620} ${Common_T03630} ${Common_T03640} ${Common_T03650} ${Common_T03660} ${Common_T03670} ${Common_T03680} ${Common_T03690} ${Common_T03700}

${Common_T03800}

${Common_T03810}

${Common_T03820}

${Common_T03830}

${Common_T03840}

- Simon Bennetts
- Tom Brennan
- Fabio Cerullo
- Oana Cornea
- Johanna Curiel
- Todd Dahl
- Luis Enriquez
- Ken Ferris
- Darío De Filippis
- Sebastien Gioria
- Tobias Gondrom
- Timo Goosen
- Anthony Harrison
- John Herrlin
- Jerry Hoff
- Marios Kourtesis
- Antonis Manaras
- Jim Manico
- Mark Miller
- Cam Morris
- Susana Romaniz
- Ravishankar Sahadevan
- Tao Sauvage
- Stephen de Vries
- Colin Watson

- ${Common_T03850}
- ${Common_T03860}
- ${Common_T03870}
- ${Common_T03880}

${Common_T03900}

${Common_T03910}

- ${Common_T03920}
  https://www.youtube.com/watch?v=i5Y0akWj31k
- ${Common_T03930}
  http://trustedsoftwarealliance.com/2014/03/21/the-owasp-cornucopia-project-with-colin-watson/
- ${Common_T03940}
  https://www.youtube.com/watch?v=Q_LE-8xNXVk

${Common_T03950}