CONHECIMENTOS ESPECÍFICOS

A respeito de redes de computadores, julgue os itens subsequentes.

61 Na topologia em anel, cada bite se propaga de modo independente, sem esperar pelo restante do pacote ao qual pertence, sendo possível que um bite percorra todo o anel enquanto outros bites são enviados ou, muitas vezes, até mesmo antes de o pacote ter sido inteiramente transmitido.

62 O FTP (*File Transfer Protocol*) é um protocolo da camada de aplicação do TCP/IP que utiliza duas conexões TCP paralelas para transferir um arquivo: uma de controle e outra de dados.

63 As trocas de mensagens entre os componentes de *hardware* ou de *software* de dispositivos conectados em rede, como, por exemplo, *smartphones* e *tablets*, são definidas por meio de protocolos, que, em última instância, envolvem as atividades de duas ou mais entidades remotas comunicantes na Internet.

64 Na comutação de circuitos, diferentemente do que ocorre na comutação de pacotes, para que haja comunicação entre os sistemas finais, os recursos necessários (como *buffers* e taxa de transmissão de enlaces) são reservados pelo período da sessão de comunicação entre os sistemas.

Julgue os itens a seguir, a respeito de endereçamento, modelo OSI e WPA2.

65 A camada física do modelo OSI, apesar de não impedir que um transmissor rápido envie uma quantidade excessiva de dados a um receptor lento, tem a função de transformar um canal de transmissão bruta em uma linha que pareça livre de erros de transmissão não detectados para a camada de enlace.

66 O WPA2 incorpora todos os aspectos da especificação de segurança para WLAN conhecida como IEEE 802.11i, ainda que não proveja serviço de autenticação na troca de mensagens entre um usuário e um AS (*authentication server*).

67 No caso de um endereçamento IPv4, as configurações para rede 192.168.1.1/24 permitem atribuir 1.024 IPs e utilizar, de fato, apenas 1.022 IPs.

Acerca de prevenção e tratamento de incidentes, julgue os itens seguintes.

68 No caso de um ataque de DoS (*Denial of Service*) a uma rede de computadores, seria mais indicado como resposta reconfigurar o roteador para minimizar efeitos de *flooding* que duplicar a configuração dos ativos envolvidos para investigação forense.

69 Suponha que os dados a seguir constituam parte de um *log* de auditoria obtido a partir da execução do comando `netstat-an`. Considere, ainda, que o estado da conexão exibido na última linha se repita de maneira contínua. Nessa situação, deve-se concluir que o ativo em questão está sofrendo um ataque DDoS (*Distributed Denial of Service*).

```
TCP  192.168.2.104:00  216.35.50.65:60973 TIME_WAIT
TCP  192.168.2.104:00  216.35.50.65:60974 TIME_WAIT
TCP  192.168.2.104:00  216.35.50.65:60975 TIME_WAIT
TCP  192.168.2.104:00  216.35.50.65:60976 TIME_WAIT
TCP  192.168.2.104:00  216.35.50.65:60977 TIME_WAIT
TCP  192.168.2.104:00  216.35.50.65:60978 TIME_WAIT
TCP  192.168.2.104:00  216.35.50.65:60979 TIME_WAIT
TCP  192.168.2.104:00  216.35.50.65:60980 TIME_WAIT
TCP  192.168.2.104:00  216.35.50.65:60981 TIME_WAIT
TCP  192.168.2.104:00  216.35.50.65:60982 TIME_WAIT
TCP  192.168.2.104:00  216.35.50.65:60983 TIME_WAIT
TCP  192.168.2.104:00  216.35.50.65:60984 TIME_WAIT
```

70 Um IDS (*Intrusion Detection System*) pode ser usado para detectar varreduras de porta e de pilha TCP, além de ataques de DoS, de inundação de largura de banda, de *worms* e de vírus.

71 Filtros de pacotes tradicionais são considerados *firewall* porque podem executar uma política de filtragem com base na combinação de endereços e números de porta, examinando cada datagrama e determinando, a partir de regras específicas, se ele deve passar ou ficar.

72 **Situação hipotética**: Um sistema apresentava falhas ao executar uma função específica X. Após a aplicação de *patch* do fabricante do *software*, o erro que causava essas falhas foi corrigido, mas outro erro na aplicação foi encontrado na função Y. **Assertiva**: Nessa situação, o erro na função Y pode ter sido causado pelo *patch*, já que essa ferramenta é capaz de alterar diversos aspectos do *software* ao qual é aplicada.

73 Um dos princípios que norteiam o gerenciamento de privilégios administrativos é o privilégio mínimo, de acordo com o qual as políticas de controle de acesso devem ser as fechadas, ou seja, somente os acessos especificamente autorizados são permitidos.

74 Os *honeypots* (potes de mel), enquanto tecnologia de detecção de intrusão, podem ser utilizados para atingir os objetivos de atrair um atacante potencial e afastá-lo de sistemas críticos e de incentivar o atacante a ficar no sistema por período de tempo suficiente para que haja resposta dos administradores, mas não para coletar informações sobre a atividade do atacante, uma vez que não foram projetados para esse fim.

75 Com a implementação do *whitelisting*, recurso que é o contrário do *blacklisting*, impossibilita-se a entrada de qualquer executável que não esteja cadastrado, dada a existência de uma lista de *emails*, domínios ou endereços IP previamente aprovados e, normalmente, não submetidos aos filtros.

Acerca dos procedimentos da análise forense digital e cadeia de custódia, julgue o item a seguir.

76 Em análise forense digital, o início da cadeia de custódia ocorre a partir da duplicação pericial do vestígio previamente adquirido.

A respeito de sistemas de arquivos, duplicação e recuperação de dados apagados ou corrompidos, julgue os próximos itens.

77 Na duplicação de um disco rígido para fins forenses, são copiados os dados regulares presentes, os arquivos apagados, os fragmentos remanescentes de arquivos e os dados que eventualmente se encontrem armazenados no espaço localizado fora dos limites das partições.

78 As técnicas de *data carving* objetivam a recuperação de dados apagados a partir da análise de dados brutos à procura de assinaturas e outras marcações. Havendo sucesso nessa busca, *data carving* realiza a recuperação de arquivos inteiros e de seus metadados, e, em alguns casos, de fragmentos de arquivos que podem ter utilidade forense.

79 No sistema de arquivos NTFS, uma cópia de segurança da MFT (*master file table*) é mantida para a recuperação em caso de eventual perda de dados da tabela. O *cluster* desse becape está no *offset* 0×38 (hexadecimal) do setor de *boot* do NTFS.

Com relação à análise de linha do tempo e à aquisição de dados em memória, julgue os seguintes itens.

80 A extração de uma imagem da memória de um sistema, conhecida como *dump* de memória, permite identificar os processos que estavam em execução no momento da extração, bem como os arquivos, as bibliotecas, chaves de registro ou *sockets* em uso por cada processo.

81 A análise de linha do tempo de eventos de interesse forense requer a existência sistematizada de registros de *logs* dos sistemas periciados para ser realizada, sendo sua aplicação limitada à análise forense de sistemas corporativos que dispõem desses recursos.

Julgue os itens seguintes, a respeito da análise de artefatos maliciosos.

82 **Situação hipotética**: Ao se carregar, em um editor de hexadecimal, um arquivo executável de nome `file.exe`, obteve-se o código `0x5a4d` no início do arquivo. Em seguida, o arquivo foi renomeado para `file.txt` e novamente carregado no editor, obtendo-se o mesmo código `0x5a4d`. **Assertiva**: Nessa situação, o código em questão se refere ao *magic number*, o qual compõe a estrutura do arquivo executável e não se altera mesmo mudando-se a extensão do arquivo, constituindo uma das formas de o sistema operacional reconhecer o tipo de arquivo.

83 Tendo como referência os códigos I e II a seguir, é correto afirmar que, no código I, foi realizada *obfuscação*, ou ofuscação, que tem, entre outros objetivos, o de tornar o código mais difícil de ser lido mediante a utilização de técnicas como mudar nomes de variáveis.

código I

```
public ExampleUI()
    {
        this.InitializeComponent();
            this.displayText.Text = new ClassX("Some
        Text").get_DisplayText();
    }
```

código II

```
public A()
    {
        this.A();
        this.a.Text = new A.A("Some Text").A();
    }
```

84 Executar com sucesso o *disassembly* não é um problema simples de resolver haja vista que sequências de código executável podem ter várias representações — algumas que podem ser inválidas — e, ao final, pode-se causar erros na funcionalidade real do programa.

85 **Situação hipotética**: Na realização de um monitoramento com a utilização do Internet Explorer, observou-se que:

I ao iniciar uma sessão em determinado sítio, um processo no sistema operacional foi iniciado, faltando uma DLL para a aplicação X;
II o processo ligado à sessão foi iniciado na pasta `c:\xyz\`;
III ainda com a sessão ativa, observou-se que a DLL foicarregada a partir da pasta `c:\xyz\` e a sessão foi interrompida;
IV no reinício do sistema operacional, a aplicação X foi carregada com a DLL em `%SystemRoot%/System32`, a mesma que se encontrava na pasta `c:\xyz\`.

Assertiva: A situação hipotética descreve um ataque DLL *hijacking* de acordo com III e IV, principalmente pela interrupção, entretanto, é descaracterizado, pois vai de encontro ao que foi descrito em I e II, porque a DLL precisa estar presente antes do início da sessão e, se foi carregada em `%SystemRoot%/System32`, é confiável e imune a ataques desse tipo.

86 **Situação hipotética**: Um analista com acesso legalmente constituído extraiu dados de dois sistemas protegidos, aos quais tinha autorização válida e vigente. Após obter os dados, o servidor os compartilhou com terceiros não autorizados, propositalmente. **Assertiva**: Essa situação descreve uma *exfiltração* de dados, ainda que o agente causador tenha autorização e fosse confiável, isso porque ela pode ocorrer da forma descrita ou ainda por meio de processo automatizado conduzido por meio de um programa malicioso.

Julgue os itens a seguir, relativos a injeção de código, engenharia reversa e *exfiltração* (ou desinfiltração) de dados.

87 **Situação hipotética**: Para um programa, foram realizadas duas engenharias reversas, com técnicas distintas, ambas a partir do programa executável. Na primeira, obteve-se o código-fonte em linguagem Assembly; na segunda, obteve-se o código-fonte na linguagem C, na qual o *software* foi originalmente desenvolvido. **Assertiva**: Essa situação descreve as técnicas de engenharia reversa conhecidas, respectivamente, como análise de propriedades estáticas e comportamento dinâmico.

88 **Situação hipotética**: Na coleta de informações de um sistema atacado pelo *malware* Y, observou-se que as chamadas às APIs do Windows estavam sendo redirecionadas para o *software* de monitoramento antes que o código da API fosse efetivamente chamado, criando informações sobre a sequência das operações do sistema executadas pela amostra de *malware*. **Assertiva**: Essa situação descreve um ataque do tipo API *hooking*, cuja característica é a garantia de que o comportamento do nível do sistema (que, em algum momento no tempo, deve usar uma chamada de API) não é ignorado, a menos que a chamada da API correspondente não seja conectada.

89 **Situação hipotética**: Uma mudança maliciosa da chamada ao sistema *web* ocorreu por meio de substituição de valor de uma variável e inserção de outra, conforme a manipulação de URL a seguir.

de http://www.site.com.br/script?variavel=X
para http://www.site.com.br/script?variavel=ABC&variavel_2=123

Assertiva: Essa situação descreve respectivamente um ataque por meio da técnica de persistência — em que há mudança no valor dos parâmetros ou variáveis — e da técnica de lateralidade — em que há a inserção de outras variáveis e(ou) parâmetros.

A respeito de inteligência de ameaças em fontes abertas (OSINT), julgue os itens a seguir.

90 OSINT é potencialmente uma fonte de informação rápida e economicamente viável, e a informação e a inteligência derivadas de OSINT podem ser potencialmente compartilhadas.

91 Informações obtidas por meio de OSINT são menos confiáveis e menos precisas que aquelas obtidas usando-se disciplinas de inteligência tradicionais.

Acerca de indicadores de comprometimento (IOC), julgue os próximos itens.

92 Uma das características de TAXII é a disponibilização de métodos de autenticação, autorização e acesso separados para produtores e consumidores, o que permite a proteção de informações relacionadas a ameaças cibernéticas que se desejem manter privadas.

93 Características técnicas que definem ameaças cibernéticas, metodologias de ataques e artefatos consequentes de intrusões são exemplos de informações compartilhadas como indicadores de comprometimento pelo *framework* OpenIOC.

94 CybOX é uma linguagem padronizada para codificação e comunicação de informações direcionada a eventos cibernéticos específicos ou casos de uso únicos de segurança cibernética para os quais são registradas observações simples, individualizadas e rigidamente estruturadas.

A respeito de estruturas de comando e controle de artefatos maliciosos, julgue os itens subsequentes.

95 A estrutura de comando e controle do *malware* Taidoor é conectada pelas máquinas comprometidas por meio de uma DLL camuflada em dados aparentemente aleatórios de um *post* de um blogue do Yahoo, criptografados por RC4, codificados em base64 e baixados por um *malware* auxiliar.

96 Uma das características das estruturas de comando e controle de *malware* descentralizadas é a garantia da existência de uma quantidade significativa de nós redundantes que devem ser atacados para que a rede de comando e controle associada ao *malware* seja desativada.

Com relação ao redirecionamento de tráfego malicioso, julgue o item seguinte.

97 As principais atribuições de um servidor DNS *sinkhole* incluem detectar e analisar comunicações maliciosas vindas da Internet e direcionadas a servidores corporativos legítimos, redirecionar esse tráfego e enviar regras para bloqueio prévio dos endereços IP nos *firewalls* corporativos.

Acerca das ameaças persistentes avançadas (APT), vulnerabilidades zero *day* e engenharia social, julgue os itens a seguir.

98 Um *exploit* elaborado para um ataque direcionado e com base em uma vulnerabilidade zero *day* permanece efetivo até que a vulnerabilidade seja publicamente revelada e a correção de *software* seja produzida, distribuída e aplicada.

99 O uso de engenharia social e o envio de mensagens contendo *links* para *websites* hospedeiros de código malicioso a fim de explorar vulnerabilidades zero *day* para pessoas cuidadosamente selecionadas e conectadas a redes corporativas são maneiras comuns de iniciar ataques de APT.

A respeito dos ataques distribuídos de negação de serviço (DDoS), julgue o próximo item.

100 Os testes CAPTCHA são eficientes contra os ataques DDoS volumétricos e os de exaustão das conexões TCP, mas são inócuos contra os ataques DDoS de camada de aplicação.

A respeito de artefatos maliciosos, julgue os itens que se seguem.

101 *Rootkits* executando em modo usuário são difíceis de detectar, possuem acesso irrestrito ao *hardware* e operam tipicamente subvertendo o fluxo de execução, ocultando-se e substituindo por código malicioso as chamadas do sistema que manipulam arquivos, processos, memória, controle de acesso e comunicações de rede.

102 O *spyware* Flame, também conhecido por sKyWIper, foi descrito como um dos mais complexos códigos maliciosos já descobertos, sendo a ele atribuída a capacidade de coletar informações da máquina infectada por meio de teclado, tela, microfone, dispositivos de armazenamento, rede, *wi-fi*, *Bluetooth*, USB e processos do sistema.

103 O instalador do *ransomware* WannaCry executa dois componentes: o primeiro usa o *exploit* EternalBlue para explorar uma vulnerabilidade do RDP (*remote desktop protocol*) do Windows e propagar o *malware*; o segundo é um componente de criptografia.

Com relação a *botnets* e *phishing*, julgue os itens a seguir.

104 Um dos motivos para o deslocamento das ocorrências de *phishing* para as redes sociais é a usabilidade dessas redes como canais de comunicação legítimos e familiares aos usuários, o que torna mais difícil distinguir mensagens de *phishing* de mensagens genuínas.

105 A atual geração de dispositivos IOT (Internet das coisas) não foi concebida com foco em segurança do *software*, o que os torna candidatos prováveis a integrar gigantescas *botnets* que, entre outras atividades rentáveis, podem ser usadas para acelerar quebras de senhas para invadir contas *online*, minerar *bitcoins* e realizar ataques de negação de serviço sob encomenda.

Com o objetivo de direcionar testes de penetração a ser executados em uma organização, um analista deve considerar os seguintes requisitos.

I Devem ser realizados ataques sem que o testador tenha conhecimento prévio acerca da infraestrutura e(ou) aplicação.

II Devem ser enviadas ao testador informações parciais e(ou) limitadas sobre os detalhes internos do programa de um sistema, simulando, por exemplo, um ataque de *hacker* externo.

Tendo como referência a situação hipotética apresentada, julgue os itens que se seguem.

106 O requisito II é uma descrição do teste de penetração do tipo *white-box*, que é normalmente considerado uma simulação de ataque por fonte interna e(ou) usuário privilegiado.

107 O requisito I é uma descrição do teste de penetração do tipo *black-box*, que pode ser realizado com ferramentas de descoberta de vulnerabilidade para a obtenção das informações iniciais sobre o sistema e a organização de fontes públicas.

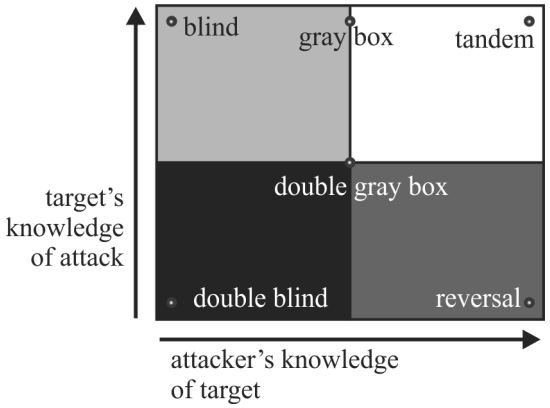
Acerca de testes de penetração, julgue os itens seguintes.

108 **Situação hipotética**: O acesso a uma aplicação *web* com permissão de administrador é realizado por meio do valor informado em uma variável, conforme a seguir.

`http://www.site.com.br/aplicacacao?profile=ascs23f8g7por04`

Assertiva: Nesse caso, de acordo com a OWASP (*Open Web Application Security Project*), o teste de penetração *black-box* automatizado é efetivo para encontrar uma vulnerabilidade, dados o valor fixo para a variável e a forma de passagem: pedido via `GET`.

109 Na situação apresentada na figura a seguir, de acordo com o OSSTMM (*The Open Source Security Testing Methodology Manual*), seria correto classificar o teste de penetração como um teste do tipo *tandem*, mas não como um teste *double blind*, haja vista que, na técnica *tandem*, o alvo não recebe notificação prévia referente ao alcance da auditoria, aos canais testados e aos vetores de teste.



110 Na análise de vulnerabilidades, uma das fases da execução do teste de penetração de acordo com o PTES (*Penetration Testing Execution Standard*), a varredura de porta é uma técnica que ajuda a obter uma visão geral básica do que pode estar disponível na rede de destino ou no *host*; na exploração, outra fase da execução de tal teste, o *fuzzing* visa recriar um protocolo ou aplicativo e enviar dados no aplicativo com o intuito de identificar uma vulnerabilidade.

111 Tcpdump é um aplicativo que recupera o conteúdo dos pacotes em uma interface de rede e permite, entre outras ações, o armazenamento dos dados do pacote em arquivo para análise posterior e a interrupção da comunicação entre emissor e receptor por meio de envio de *kill* (*-k*).

Julgue os itens a seguir, a respeito da identificação de condições de erro.

112 **Situação hipotética**: Um servidor de banco de dados para utilização de *web services* foi configurado com a porta padrão indicada pelo fabricante, sendo somente por meio dessa porta que as aplicações farão acesso ao servidor. **Assertiva**: Nessa situação, se a porta de acesso ao banco de dados for alterada, recomenda-se não atualizar as aplicações, para garantir maior segurança dos dados trafegados.

113 **Situação hipotética**: A administração de um sítio — mudança do conteúdo e alteração dos arquivos — é realizada por meio de console. No primeiro acesso, embora seja solicitado que o administrador altere a senha inicial, que é padrão para todos os clientes, ele pode manter a senha padrão para administração do sítio por tempo indeterminado. **Assertiva**: Nessa situação, é indicado o envio, para o administrador, de uma senha específica e aleatória, válida por tempo determinado, e que deve ser obrigatoriamente alterada no primeiro acesso.

114 Na instalação de um servidor Apache, caso as classes Java compiladas fiquem em um dos diretórios padrões instalados, um ataque não conseguirá ver o código e nem terá acesso às regras de negócios encontradas nessas classes.

Devido ao baixo custo, o *fuzzing* é bastante utilizado pelas empresas de segurança e *hackers*, para testar aplicações *web* e listar suas vulnerabilidades. A esse respeito, julgue os itens a seguir.

115 Os *fuzzers black-box* de aplicações *web*, por questão de segurança, não permitem requisições que mostrem os valores de resposta na URL, o que impede a avaliação das respostas retornadas pelo servidor por meio de expressões regulares ou de funções *hash*, sem o conhecimento prévio dos valores da resposta.

116 Na técnica conhecida como *fuzzy white-box*, a equipe de teste possui acesso ao código fonte da aplicação no servidor local e consegue executar os testes *fuzzing* por meio de algoritmos com casos de teste gerando resultado mais rápido e preciso para o gestor.

Os itens a seguir apresentam uma situação hipotética seguida de uma assertiva a ser julgada a respeito de identificação de vulnerabilidades por inspeção de código.

117 Um arquivo de configuração é acessado por uma classe Java que guarda, em um objeto em memória, o acesso ao `dataSource` do banco de dados do servidor. Quando compilado, esse arquivo é criptografado. Nesse caso, para evitar alteração ou modificação do arquivo por terceiros, uma solução seria guardar o arquivo no repositório GIT, de forma privada.

118 A arquitetura dos sistemas de uma organização utiliza uma mesma base de dados, ou seja, todos os sistemas acessam e gravam no mesmo banco de dados. Nessa arquitetura, todos os dados de acesso, como *login* e senha, estão armazenados em um arquivo `.txt` de configuração padrão no repositório central. Nessa situação, visando diminuir a fragilidade da arquitetura, é indicado que todos os sistemas tenham um `dataSource` específico de acesso aos seus dados.

Considerando o acesso a um sítio de uma empresa, desenvolvido em arquitetura Web/PHP, julgue os itens a seguir, a respeito de segurança de aplicações *web*.

119 Na situação de um URL/PHP que receba *upload* de arquivos, a maneira mais indicada de evitar um ataque de inclusão de arquivo é validar a URL que está subindo o arquivo junto com um *token*.

120 **Situação hipotética**: Navegando-se pelo sítio, descobriu-se o `test=Query` por meio de um método `GET` na URL, a qual foi usada para pesquisar o banco de dados e suas colunas, utilizando-se a ferramenta SQLMap. **Assertiva**: Essa situação ilustra uma forma de se explorar a fragilidade do sítio por meio de SQL *injection*.

121 **Situação hipotética**: Um empregado da empresa que deveria ter acesso apenas a seu contracheque, ao inspecionar o código, observou que é possível alterar os seus dados `GET` de busca e acessar o contracheque de outro empregado, do qual conhece o *user*, que é o filtro do sistema. **Assertiva**: Essa situação ilustra um problema de *cross-site scripting* (XSS), que pode ser resolvido alterando-se o método do formulário para `post`.

No que se refere à vulnerabilidade em navegadores *web*, julgue os seguintes itens.

122 No ambiente Windows 10, a opção de atualização automática não está disponível para o Edge, então, para que o navegador seja atualizado, é necessário solicitação do administrador de redes.

123 **Situação hipotética**: Após a instalação de um *plugin* do navegador, um usuário, ao tentar acessar sua conta bancária *online*, verificou que a URL do banco tinha sido modificada e o acesso estava sendo direcionado para outro domínio; verificou também que arquivos do sistema Windows tinham sido modificados. **Assertiva**: Essa situação ilustra um problema que pode ser resolvido alterando-se a segurança do navegador para máxima, sem a necessidade de atualização do antivírus.

124 Para permitir a correção automática de eventuais falhas de segurança encontradas no Google Chrome, é necessário que o administrador libere o *download* das atualizações do navegador.

Acerca do armazenamento de dados na plataforma Android, julgue os seguintes itens.

125 O Android disponibiliza um banco de dados público local, orientado a objetos, para o armazenamento de dados estruturados, o que possibilita o gerenciamento das aplicações e dos dados de forma rápida e segura.

126 Optando-se por *external storage*, os dados de uma aplicação serão armazenados no cartão de memória do celular, o que permite espaço extra para o aplicativo, além de acesso rápido aos dados da App.

Julgue os próximos itens, relativos ao sistema operacional Android.

127 Mesmo controlando o *login* e a senha do usuário via contas Google, um aplicativo pode capturar e enviar arquivos armazenados no cartão SD do celular que utiliza o sistema Android.

128 Quando a Internet está disponível, os aplicativos executados em segundo plano podem efetuar requisições, que utilizam muita carga da bateria e podem ocasionar erros nos aplicativos, por isso, na versão 8.0 do sistema, os manifestos não podem ocorrer para transmissões implícitas.

129 Para garantir que o *software* gerado no servidor chegue ao usuário final, utiliza-se um certificado *code signing*, que altera o *software* e também insere uma assinatura do desenvolvedor ou fabricante.

Julgue os itens a seguir, em relação à vulnerabilidade dos dispositivos móveis ou navegadores *web* usados para acessar a Internet.

130 Por meio de um *keylogger* inserido em uma *app* maliciosa instalada no dispositivo móvel, é possível a captura das teclas digitadas pelo usuário quando da utilização de navegadores *web*.

131 **Situação hipotética**: Para efetuar uma compra *online*, um usuário permitiu a instalação de um *plugin* no seu navegador *web*. Dias depois, ele constatou que seu endereço de entrega tinha sido alterado e novas compras tinham sido efetuadas no sítio. **Assertiva**: Essa situação ilustra um tipo de roubo de sessão de usuário que permanece *logado* no sítio, o qual usa *cookies* para manter a sessão; nesse caso, o invasor pode editar o *cookie* da sessão válida para alterar os dados e efetuar compras, como se fosse o verdadeiro usuário.

132 Ainda que o usuário de dispositivo Android instale *apps* de lojas não oficiais do Google, o dispositivo não será afetado por *trojans*, devido a sua arquitetura baseada em Unix, que são códigos que abrem portas de manipulação remota do celular.

Com relação a criptografia simétrica e assimétrica e a ataques a sistemas de criptografia, julgue os itens a seguir.

133 Na criptografia assimétrica, as duas partes comunicantes compartilham a mesma chave, que precisa ser protegida contra acesso por outras partes.

134 Um ataque de criptoanálise linear tem como essência a observação do comportamento de pares de blocos de texto em evolução a cada rodada da cifra, em vez da observação da evolução de um único bloco de texto.

135 Para se conseguir sucesso em ataques por força bruta, em média, um terço de todas as chaves possíveis precisa ser experimentada.

Acerca de assinatura digital, certificação digital e infraestruturas de chaves públicas, julgue os itens a seguir.

136 Uma infraestrutura de chaves públicas é o conjunto de *hardware*, *software*, pessoas, políticas e processos necessários para administrar todo o ciclo de vida de certificados digitais desenvolvidos com base em criptografia assimétrica.

137 Uma assinatura digital direta é formada criptografando-se a mensagem inteira, ou um código de *hash* da mensagem, com a chave privada do emissor da mensagem.

138 Qualquer usuário pode verificar a autenticidade da entidade que gerou um certificado digital, mas apenas usuários autorizados e credenciados podem consultar as informações e a chave pública do proprietário do certificado.

Julgue os itens seguintes, a respeito de controlador lógico programável (CLP).

139 Na função NA (normalmente aberto) da linguagem Ladder, o estado da saída digital do CLP será exatamente o mesmo observado na entrada digital.

140 Para que o programa monitor de um CLP tenha funcionamento correto e seguro, ele exige memória RAM exclusiva.

141 O CLP com entradas analógicas é normalmente utilizado em processos que exijam controle de alta precisão.

Em relação a sistemas ICS/SCADA, julgue os itens a seguir.

142 O emprego de protocolos é um fator positivo em relação à segurança de sistemas SCADA, já que, embora proprietários, eles são de conhecimento público e, portanto, amplamente avaliados quanto à segurança.

143 Na configuração típica de um sistema ICS/SCADA, a utilização de um CLP exige a sua ligação a uma unidade terminal remota.

144 Em sistemas SCADA, o uso de comunicação por meio de satélite tem aumentado como resposta a fatores de segurança.

Acerca de inteligência cibernética, julgue os itens a seguir.

145 O uso de domínios *web* de final `.on` e de roteadores em formato de *proxy* são características da *dark web*.

146 O aplicativo TOR permite o acesso a sítios na *deep web*, isto é, sítios que não possuem conteúdo disponibilizado em mecanismos de busca.

147 O registro e a análise de conjuntos de dados referentes a eventos de segurança da informação são úteis para a identificação de anomalias; esse tipo de recurso pode ser provido com uma solução de *big data*.

148 A utilização da Internet para o recrutamento de jovens pelos grupos radicais que praticam atos terroristas é considerada como terrorismo cibernético.

A respeito de Internet das coisas (IoT), julgue os itens que se seguem.

149 Em uma residência, caracteriza uma solução de IoT a instalação de um detector de fumaças capaz de gerar alertas em caso de fumaça e ser acionado, a partir de um *smartphone*, para iniciar um mecanismo de reação.

150 Redes *wi-fi* e *Bluetooth* podem ser utilizadas para IoT, já NFC (*near field communication*) não atende a demandas de IoT.

Espaço livre