Segurança em Sistemas de Informação

Aula 6

Luis Gonzaga de Paulo

Conversa Inicial

A segurança dos sistemas e do software em geral é um componente fundamental para a segurança da informação, sendo o item de maior atenção no ambiente de tecnologia da informação. Isso não acontece sem motivo, pois, na Era da informação, da automação e da conectividade, são os sistemas e os softwares os guardiões primeiros da informação, e a dependência por parte dos seres humanos em relação a eles só tem aumentado com o passar do tempo.

Contextualizando

Nos dias atuais, a segurança da informação é totalmente dependente da segurança dos sistemas e do software em geral. Nossas vidas e praticamente tudo o que fazemos e utilizamos dependem de informações geradas, armazenadas, processadas ou transportadas por algum tipo de software ou sistema. Essa dependência justifica o máximo cuidado com a segurança desses softwares e dos sistemas como forma de preservar a segurança da informação por eles manipulada. Nesta aula, vamos abordar diversos aspectos dessa necessidade de segurança em todo o ciclo de vida do software e dos sistemas, bem como em ambientes computacionais emergentes, a computação móvel e a internet das coisas.

Pesquise

Antes de iniciamos, que tal pesquisar um padrão de avaliação da segurança do software, a ISO 15408?

Tema 1: Desenvolvimento e Testes

É certo que significativa parcela dos problemas enfrentados por softwares de qualquer natureza decorre das falhas humanas, quer seja no processo de produção do software, quer seja em sua configuração e adequação ao ambiente, quer seja em seu uso. Os diversos métodos, técnicas,

ferramentas e abordagens procuram mitigar os riscos inerentes a esse processo e decorrentes dessa premissa. Sua constante evolução, além de buscar maior eficiência em termos de custo e prazo para a produção de software, também objetiva tornar o processo de desenvolvimento de software mais padronizado e seguro. Dessa forma, procura-se aumentar a confiabilidade do produto final e reduzir sua dependência do fator humano no que tange à qualidade e à segurança.

Devido à exposição do software aos problemas das diversas naturezas citadas, é imprescindível que haja, no processo de desenvolvimento de software, um planejamento robusto e abrangente, que considere os riscos na proporção adequada à segurança exigida para o software a ser produzido. Também é necessário haver especial atenção aos procedimentos que garantirão a qualidade do software – e de forma especial aos testes a serem realizados durante todo o processo –, bem como à padronização e ao uso de métodos, técnicas e ferramentas como nível adequado de maturidade requerido pelo software, e também à automação de tarefas e atividades padronizadas, reduzindo a interferência humana e, por conseguinte, a possibilidade de geração de problemas.

Todo produto de software é um conjunto de componentes e funcionalidades que interagem entre si, com o usuário, com o ambiente computacional no qual opera – hardware, sistema operacional, redes, entre outros – e, portanto, dependente do comportamento e do funcionamento de todo esse sistema (Goertzel et al, 2011).

Em uma análise em relação aos impactos dos componentes do software sobre a segurança das pessoas e da informação, Goertzel et al (2011) inferem que qualquer anomalia no funcionamento ou no comportamento de um ou mais desses elementos pode causar problemas para o software e, consequentemente, falhas de segurança – considerada aqui não somente a segurança da informação, mas a segurança no aspecto mais amplo, que pode inclusive colocar em risco a vida e a integridade de seres vivos.

Estendendo essa dependência ao processo de desenvolvimento de software, Goertzel et al (2011) consideram que há ainda uma dependência decorrente das atividades realizadas naquele processo, desde os primeiros momentos, pois certamente são tomadas decisões e executadas ações que se tornarão cruciais para o funcionamento adequado do software. Essas decisões e ações contribuem para a qualidade do software, para sua sustentabilidade e para a segurança das informações a serem tratadas por ele, bem como para a reação adequada às falhas e o enfrentamento de ameaças.

Além disso, as próprias atividades executadas no processo de desenvolvimento de software podem inserir problemas neste, muitos deles de forma latente, os quais se manifestarão após sua disponibilização para uso. Há de se considerar também a possibilidade de medidas e contramedidas que se mostrarão inócuas ou insuficientes para conter falhas e repelir ameaças. Alguns desses aspectos são apontados na sequência.

Modelo cascata

Do ponto de vista da segurança da informação, as principais vantagens são o planejar antes de fazer, definindo prazos e resultados esperados – os artefatos entregáveis – para cada etapa, e o uso de documentação formal. Por outro lado, alterações ou mudanças não são bem-vindas – mesmo perante uma expectativa de requisitos acurados que não é realista.

O software só é avaliado de forma efetiva e total na fase de integração e testes, quando é tarde para identificar os problemas – especialmente se houve algum atraso nas etapas iniciais, o qual acaba sendo compensado nesta etapa. Há também a dificuldade de integrar o modelo ao gerenciamento de riscos, o que é especialmente complicado para projetos de maior porte (Munassar; Govardhan, 2010).

Modelo iterativo

Uma das vantagens do modelo iterativo é ele facilitar a identificação de

problemas com a especificação de requisitos e permitir correções e mudanças durante praticamente todo o projeto. Por outro lado, é muito difícil dividir um sistema – especialmente um projeto de pequeno porte – de modo a permitir iterações ou ciclos que entreguem funcionalidades completas. Também se pode considerar que o modelo iterativo é a base das metodologias ágeis. Sob o aspecto de segurança da informação, o mais crítico é que possíveis revisões, mudanças ou falhas nas integrações entre as entregas das iterações resultem na incorporação de erros ou falhas de segurança no software.

Modelo RAD – Rapid Application Development

No que tange à segurança da informação, a excessiva simplificação e a pressão por resultados são um fator crítico, podendo impactar em falhas latentes mais graves que repercutirão no software quando já em uso. A necessidade de profissionais altamente especializados pode representar um problema na medida em que restringe o conhecimento daqueles às áreas específicas, ou impacta fortemente nos custos ao prover especialistas de diversas áreas – incluindo-se a segurança da informação – para compor a equipe do projeto.

Modelo do processo unificado

No que compete à segurança da informação, o comportamento do modelo *Unified Process* é bastante similar ao RAD, porém incorporando as especificidades da orientação a objetos.

Modelo espiral

É um processo bastante natural, na medida em que há uma evolução na maturidade tanto do cliente do software quanto da equipe de projeto, reduzindo os riscos de insatisfação. Em função disso, o fato de adicionar novos requisitos ou promover mudanças somente a partir do momento em que estes estão em um elevado nível de maturidade, juntamente com o software em

desenvolvimento, representa um diferencial positivo.

Entretanto, há a necessidade não apenas de um envolvimento maior do cliente e de sua contínua evolução, como também do aprimoramento do relacionamento destes com os desenvolvedores. Esses dois aspectos tornam necessário um forte componente de gestão para evitar que o processo entre em uma espiral infinita, jamais apresentando o software por completo.

A incorporação da análise de risco a cada iteração reforça a qualidade, e o software é entregue de forma recorrente, desde muito cedo. Grandes projetos de missão crítica são bem atendidos por este modelo. Porém, a segurança da informação pode ser comprometida com as sucessivas mudanças – a gestão de mudanças torna-se um fator de risco, e o custo aumenta. A análise de riscos requer expertise e especialistas, e é fator determinante do sucesso do projeto (Munassar; Govardhan, 2010).

Teste de software

Diversos modelos de desenvolvimento de software – em especial o SDD, já estudado – visam à adequação e à efetividade do teste de segurança da informação, buscando estabelecer um ponto de equilíbrio entre a flexibilidade, a facilidade de implementação, os recursos necessários e os custos dos processos de teste abordados. Isso implica estabelecer o mais cedo possível os critérios viáveis de avaliação, usando o conhecimento das técnicas e dos métodos de testes voltados à segurança da informação como forma de prover aplicações e serviços seguros.

O início das atividades de teste deve ser o mais antecipado possível dentro do SDLC, independente do modelo e das técnicas empregadas. Os requisitos de segurança, que serão a base para a definição dos casos de teste, devem fazer parte do processo formal e ser definidos juntamente com o processo de especificação funcional do software a ser desenvolvido. A partir daí, a produção e o refinamento de casos de teste devem ter sequência a cada iteração, fase ou ciclo do SDLC, de modo que, ao chegar à etapa de testes de

homologação ou aceitação pelo usuário, tais requisitos tenham sido atendidos satisfatoriamente e a segurança implícita – que faz parte da expectativa do usuário – seja efetiva e comprovada.

Como já visto, para que isso aconteça, é necessário que se inicie o SDLC analisando o risco inerente ao uso da informação para a qual o software está sendo projetado. Dessa forma, são estabelecidos os valores de cada informação manipulada ou produzida pelo software, bem como a prioridade de tratamento da segurança da informação de acordo com esse valor. Em seguida, deverá ser analisada a exposição dessas informações ao risco, isto é, as vulnerabilidades e as ameaças que possam explorar essas vulnerabilidades. Para isso, é necessário que o desenvolvedor raciocine de modo diferente, assumindo o papel de atacante, com o intuito de explorar as vulnerabilidades do software, produzindo os casos de uso impróprio, cenários nos quais registrará suas ações para violar a segurança da informação tendo como premissa as vulnerabilidades do software e as ameaças conhecidas.

Os casos de uso impróprio serão utilizados para a modelagem dos procedimentos de resposta aos ataques – as contramedidas – a serem implementados para eliminar ou reduzir as vulnerabilidades a níveis aceitáveis, conforme definido na análise de risco. Além disso, também possibilitarão não apenas a definição dos casos de teste para a experimentação da efetividade das contramedidas estabelecidas, mas também a especificação e a construção das máquinas de ataque, que serão utilizadas para os testes de segurança e para os testes de invasão.

Os principais problemas de uma abordagem precoce do teste em geral – e do teste de segurança da informação em específico – no SDLC decorrem da pressão exercida pelos prazos e custos. Além disso, o baixo nível de automação de tarefas de teste, a geração e manipulação de massa de teste e a necessidade de manter constante atualização sobre as técnicas de ataques e as vulnerabilidades exploradas representam um esforço adicional significativo e que coloca o processo de teste em evidência quando há a necessidade de

redução de esforço, de custos ou de prazo.

A abordagem tradicional, na qual os procedimentos de teste de segurança são tratados em sua plenitude somente após os ciclos de desenvolvimento, fomenta uma prática pouco recomendável de recompor o cronograma e efetuar ajustes de esforço ou de custos reduzindo-se as atividades de teste, e assim comprometendo a qualidade do software. Além disso, uma abordagem tardia impede a implementação de um processo cíclico de teste, interagindo com o desenvolvimento e com a análise de riscos constantemente atualizada e revista, o que só é possível conseguir com uma abordagem proativa e assertiva da questão da segurança da informação no processo de desenvolvimento de software.

Como alternativa para fazer frente a essas situações, é recomendável definir a elaboração do plano de teste na sequência da análise de riscos e a elaboração do teste de segurança baseado em riscos ainda durante a concepção do software, na fase de análise e projeto, bem como a construção das máquinas de ataque em conjunto com a construção do software. Com isso, o ferramental básico para os testes de segurança é disponibilizado a tempo, reduzindo a pressão na etapa de testes propriamente dita.

Outro aspecto que visa reforçar a efetividade do teste é a automação, por exemplo por meio da utilização das máquinas de ataque que, além de fazerem parte do universo conhecido pela equipe de desenvolvimento em virtude de serem desenvolvidas por ela, também devem ter sido projetadas para a integração com ferramentas de automação de teste, como já salientado.

E, finalmente, a base de conhecimento é um importante referencial tanto para a elaboração do plano de teste e do teste de segurança em si quanto para a execução desse teste, uma vez que retrata o comportamento, os objetivos e principalmente o funcionamento das ameaças, o que permite reproduzir os ataques e os problemas de forma real.

Tema 2: Operação e Manutenção

Segurança da operação

Após a implantação do software, é necessário manter a avaliação da segurança da informação dele. Além da função de suporte operacional normalmente prestado, a continuidade da operação tem por objetivo realimentar a base de conhecimento por meio do registro de ocorrências de segurança da informação pelas equipes de suporte. Esse registro servirá para avaliar o comportamento do produto e a efetividade das contramedidas, além de sinalizar quando ocorrem novas ameaças ou mudanças no cenário de operação do software.

Também deve ser providenciado o registro de ocorrências por meio do próprio software, desde que consentido pelo usuário, encaminhando para a equipe de desenvolvimento e da segurança da informação um relato das ocorrências e exceções, de forma a permitir a análise da ocorrência, o tratamento por meio da aplicação de correções e a atualização da base de conhecimento. Essa prática já é adotada por grande parte dos fornecedores de software comercial e contribui para o melhoramento da qualidade dos produtos.

Outra forma de feedback de grande importância é o relato de problemas por parte dos próprios usuários por meio de canais de atendimento, como SACs, e-mail e redes sociais. Em especial, é possível identificar particularidades das ocorrências e também surtos epidêmicos típicos de novas ameaças.

O modelo SDD, já estudado anteriormente, inclui na segurança da operação o trabalho voltado não apenas ao recebimento, à análise, ao refinamento e à classificação das informações relativas à segurança da informação, mas também ao respectivo registro na base de conhecimento para realimentar o modelo no SDLC, possibilitando o melhoramento contínuo da segurança da informação no software. Uma possibilidade de promover esse feedback é a inclusão de uma funcionalidade de tratamento de erros que permita o registro da opinião e das considerações do usuário de forma

anônima, seja em formulário específico na internet, e-mail ou mensagem SMS/MMS.

Segurança na manutenção

A partir do momento em que um sistema ou software passa a ser utilizado, inicia-se o período de manutenção, sustentação ou continuidade operacional. É essa atividade que preserva o valor do sistema. Uma análise interessante sobre a manutenção do software foi a proposta por Meir Lehman. Ele nasceu na Alemanha e mudou-se para a Inglaterra na década de 1930, onde trabalhou na IBM entre 1964 e 1972. Em 1974, publicou o texto conhecido como as “Leis de Lehman” sobre evolução de software. Lehman considera que:

1) os sistemas evoluem e a mudança é contínua, por isso a manutenção é inexorável. Os cenários mudam, os requisitos mudam e por isso o sistema precisa ser alterado. E, uma vez alterado, o sistema também altera o ambiente ao qual pertence;

2) a complexidade dos sistemas aumenta à medida que evolui ou sofre mudanças, ao mesmo tempo em que sua estrutura original sofre uma degradação;

3) um sistema abrangente gera um meio ambiente próprio que tende à autorregulação, e isso se reflete nos seus atributos, como tamanho, taxa de erros e tempo para novas versões;

4) durante o ciclo de vida de um software, sua taxa de manutenção é quase constante, pouco refletindo mudança de recursos ou pessoal envolvido;

5) à medida que o software passa por manutenções, evolução e incrementos, aumenta também o risco de falhas, de maneira quase constante;

6) as funcionalidades do software têm de ser incrementadas continuamente para manter o grau de satisfação do usuário;

7) a menos que o sistema passe por manutenção e evolução constante, a percepção da qualidade dos resultados por parte do usuário vai

decrescendo;

8) há necessidade de um constante *feedback* dos agentes que interagem com o software para que a evolução seja proveitosa e efetiva.

Essas considerações prevalecem na atualidade, mesmo com os novos paradigmas de orientação a objeto, metodologias ágeis, software livre, *cloud computing* e Big Data. Além disso, impactam diretamente nos aspectos de segurança da informação e de sistemas. Por isso, é necessário considerá-las no processo de manutenção, bem como utilizar os modelos de governança aplicáveis, como o ITIL e o COBIT, já estudados.

Tema 3: Gestão de Configuração e de Mudanças

A gestão – ou gerenciamento – de configuração ou *Configuration Management* é parte do processo da qualidade de software baseado em padrões que têm por objetivo manter o controle e a confiabilidade do software gerado nas diversas etapas ou ciclos do desenvolvimento. Inclui a rastreabilidade das alterações e versões e o controle de mudanças, mas não se limita a eles. A gestão de configuração também se ocupa do ambiente – as mudanças que impactam no software e também as mudanças que o software exige ou provoca no ambiente.

A gestão – ou gerenciamento – de mudanças ou *Change Management* é um processo que tem por finalidade reduzir o impacto das alterações e evoluções do software, uma vez que é sabido que essas intervenções tendem a gerar problemas, falhas e interrupções.

Ambas, gestão de configuração e gestão de mudança, fazem parte do framework do ITIL e, de acordo com sua complexidade, dispõem de ferramentas de automação para o uso adequado e aderente aos padrões, como o IEEE 828-1983, a ISO 9000 e o CMMI, do SEI.

Tema 4: Software para Dispositivos Móveis

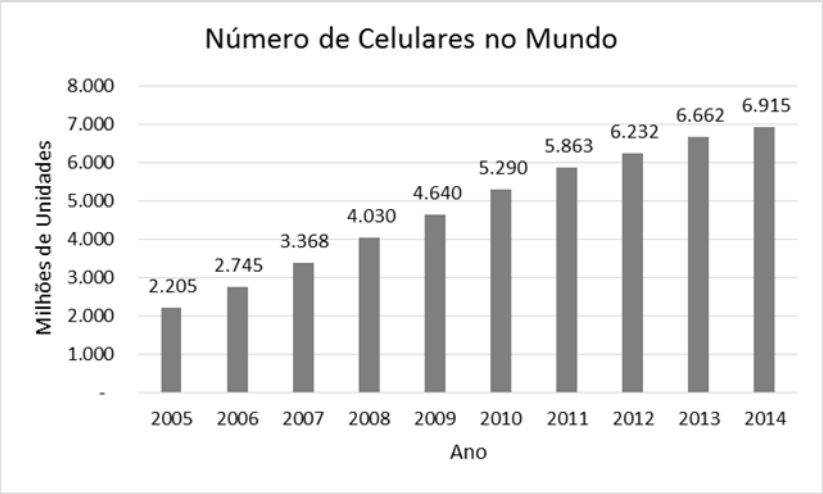
A popularização do uso de dispositivos de computação móvel – ou

mobile, principalmente *smartphones* e *tablets* – e, particularmente no Brasil, o aumento do uso dos serviços de comunicação de dados decorrente da redução de custo – quer seja dos equipamentos quer seja dos serviços – e o aumento da velocidade de acesso disponibilizada pelas prestadoras de serviços de comunicação com as redes 3G e 4G alimentam o mercado, estimulando o crescimento do número de aplicações disponíveis.

Os desenvolvedores de software para esses dispositivos devem adotar procedimentos de segurança no desenvolvimento de software, reduzindo sua vulnerabilidade, especialmente em função de tendências como o BYOD – *Bring Your Own Device* –, literalmente, traga seu próprio dispositivo – e a IOT – *Internet of Things* – Internet das Coisas – apresentarem-se como diferencial competitivo e potencial para viabilizar a comunicação universal e integrada. Por isso, é necessário reforçar a segurança da informação nos dispositivos móveis e garantir a confiabilidade do software nesses dispositivos, pois tais tendências são irreversíveis (Taurion, 2013).

Os dispositivos de computação móvel – em especial os *tablets* e os *smartphones*, têm experimentado um grande crescimento tanto em número de equipamentos, como mostrado na Figura 3, quanto no de aplicações, mas também em ataques que comprometem a segurança da informação de seus usuários (La Polla et al, 2013). A expansão dos recursos de comunicação à disposição desses dispositivos, tais como redes 3G, 4G e 5G, Wi-Fi, Bluetooth e NFC, aliada à aplicação cada vez mais intensa em processos de negócio das empresas ou em atividades pessoais que tratam informações de valor, vem tornando esse ambiente um alvo muito visado pelos ataques.

Figura 1 - Crescimento do número de celulares no mundo



Fonte: Adaptado de UIT, 2014

Esse crescimento é notado em todos os segmentos do uso dos dispositivos, bem como nos diversos sistemas operacionais, conforme mostrado na Tabela 1, mas acentua-se no caso do Android, sistema operacional distribuído pela Google, uma vez que ele equipa mais que dois terços dos dispositivos comercializados nos últimos cinco anos (Gartner, 2014).

Figura 2 - Vendas de Equipamentos por Sistema Operacional

Sistema Operacional	3T14		3T13	
	Unidades X 1000	% Mercado	Unidades X 1000	% Mercado
Android	250.060	83,07%	250.243	84,74%
iOS	38.186	12,69%	30.330	10,27%
Windows	9.033	3,00%	8.916	3,02%
BlackBerry	2.419	0,80%	4.401	1,49%
Outros	1.310	0,44%	1.407	0,48%
TOTAL	**301.008**	**100%**	**295.297**	**100%**

Fonte: Adaptado de Gartner, 2014

Comparado ao número de equipamentos, apontado pela UIT (2014) em cerca de sete bilhões ao final de 2014, e ao número de usuários, a quantidade de malwares não é significativa, porém vem crescendo de forma a comprometer os avanços obtidos em termos de aplicação desses dispositivos. Boa parte da infraestrutura tecnológica empregada pelos dispositivos móveis

não foi inicialmente projetada para suportar tamanho crescimento nem tampouco o intenso uso para o tráfego de dados – como no caso das opções disponíveis para a conectividade, tais como GSM, GRPS/EDGE, UMTS, HSPA, Bluetooth, NFC e até o próprio IEEE 802.11. Tampouco os sistemas operacionais e as aplicações desses dispositivos, limitados pela capacidade de processamento, armazenamento e disponibilidade de serviços.

Dessa forma, o cenário atual requer uma criteriosa avaliação com o intuito de prover condições para os usuários desses equipamentos e para os desenvolvedores de software que lhes permitam fazer uso de forma segura e efetiva de seus recursos, seja para o lazer ou para uso profissional, para acesso on-line a bancos ou compras na internet.

Em um ambiente de dispositivos móveis, diversos atores contribuem para a mitigação dos riscos relativos à segurança da informação (La Polla et al, 2013). Os fabricantes de hardware, que às vezes fornecem também o sistema operacional e bibliotecas para desenvolvimento de aplicações, respondem pela segurança física, mecânica e computacional no que diz respeito à confiabilidade do equipamento e à compatibilidade com funcionalidades, normas e regulamentos. As companhias operadoras proveem a infraestrutura de comunicação e também ofertam facilidades e serviços distintos, que dependem das funcionalidades do hardware e do sistema operacional e também estão sujeitos a normas e regulamentos bastante diversificados em virtude da atuação geográfica e da legislação. Os desenvolvedores de software utilizam as funcionalidades e facilidades para prover as aplicações, as quais são exploradas pelos usuários.

Cada um desses atores tem responsabilidades sobre a segurança da informação nesses ambientes, e podem ser alcançados com abordagens distintas quanto à prevenção a ataques e contramedidas de segurança (La Polla et al, 2013), a saber:

– Usuários devem ser informados dos possíveis problemas, orientados e educados a utilizar seus dispositivos corretamente e de modo

seguro.

– Desenvolvedores devem adotar medidas de proteção à segurança da informação, as mais atualizadas possíveis e que permeiem todo o ciclo de vida do software, tendo como referência as vulnerabilidades e os ataques já conhecidos.

– Operadoras devem reforçar os mecanismos de identificação e defesa das redes, além de adotar medidas preventivas contra os ataques.

– Fabricantes devem ser ágeis na identificação de falhas e na atualização do hardware, sistemas operacionais e bibliotecas com o intuito de reduzir ou eliminar falhas e vulnerabilidades que possam motivar ataques.

A infraestrutura da computação móvel

A tecnologia disponível para os dispositivos móveis conectarem-se às redes divide-se basicamente em duas modalidades: a rede de telefonia e as redes sem fio.

No primeiro modo, estão as tecnologias criadas para a mobilidade – porém não necessariamente para o tráfego de dados –, como GSM, GPRS, EDGE e UMTS. A principal característica dessa tecnologia é o uso para a comunicação por voz e mensagens curtas de texto, o SMS – *Short Message Service*. Os aprimoramentos da tecnologia possibilitaram a expansão do uso para mensagens multimídia, o MMS – *Multimidia Message Service*, as aplicações WAP – *Wireless Application Protocol*, e-mail, teleconferências e finalmente acesso à internet (La Polla et al, 2013).

No segundo modo, estão as tecnologias de rede local sem fio – WLAN, entre as quais se destacam o Bluetooth e o Wi-Fi (IEEE 802.11). Apesar de terem sua origem na necessidade de conexão de equipamentos móveis – como os notebooks – às redes locais, essas tecnologias não privilegiam a mobilidade, mas, sim, a conectividade com redes de tráfego de dados, alta velocidade e curta distância, no intervalo de uma dezena a poucas centenas de metros (La Polla et al, 2013).

Particularidades dos dispositivos móveis

Ao tratar de segurança da informação em ambiente de dispositivos móveis, há de se considerar que existem diferenças importantes entre esse ambiente e o ambiente tradicional de computação pessoal, o famoso PC – notebooks e desktops. Essas diferenças são primordiais na avaliação dos riscos e na disposição de soluções para os problemas de segurança da informação de tais dispositivos, pois, ao mesmo tempo em que ampliam as vulnerabilidades – e, portanto, as opções de ataque –, reduzem o espaço e os recursos para a utilização de mecanismos tradicionais de defesa. Dentre as principais diferenças, cabe citar as elencadas por La Polla et al (2013):

– **Mobilidade**: Os dispositivos móveis estão disponíveis para uso e acesso de seus usuários praticamente o tempo todo, e não somente nas residências ou ambiente de trabalho, mas também em locais de acesso público e coletivo, como meios de transporte, shoppings, instituições e órgãos públicos, entre outros. Evidentemente, essa exposição constante, aliada à falta de mecanismos de alerta e proteção adequados, favorece os ataques e a proliferação de problemas de segurança da informação.

– **Personalização**: Por se tratar de dispositivos que foram incorporados ao conjunto de acessórios tecnológicos de uso pessoal, mais especificamente por estarem associados a um número de telefone, os dispositivos móveis são submetidos a um elevado índice de personalização, exigindo uma abordagem diferenciada no que diz respeito à identificação de vulnerabilidades e aos hábitos de uso. Esse aspecto é particularmente significativo quando se trata da abordagem comportamental para a identificação de falhas e problemas, ou do estabelecimento de um padrão de comportamento para a identificação de tentativas de ataques ou fraudes.

– **Conectividade**: Os dispositivos móveis fazem uso intenso da conectividade proporcionada pela infraestrutura de comunicação para dispositivos móveis, seja pela rede de telefonia móvel celular, redes Wi-Fi, Wi-Max, Bluetooth e NFC. Além disso, tais dispositivos podem permanecer

conectados a mais de uma rede simultaneamente, funcionando como gateway, bridge ou ponto de acesso.

– **Convergência de tecnologias**: Para prover os diversos serviços oferecidos aos usuários desses dispositivos, inúmeras tecnologias são empregadas, partindo da comutação de circuitos de voz até a utilização de pacotes de dados, passando por sistemas de mensagens curtas (SMS) e multimídia (MMS), streaming de áudio e vídeo, recepção de TV digital e outras. Todas essas tecnologias são gerenciadas por um único mecanismo de hardware e pelo mesmo sistema operacional, compartilhando capacidade de processamento e armazenamento e as interfaces do dispositivo.

– **Capacidades reduzidas**: Em que pese o contínuo avanço impulsionado pela crescente demanda de recursos computacionais, os dispositivos móveis ainda dispõem de menos recursos quando comparados à computação pessoal – os PCs. A capacidade de processamento, as interfaces restritivas e a capacidade de armazenamento são reduzidas e restringem o uso de mecanismos de defesa mais aprimorados, como antivírus ou firewalls, comuns na plataforma PC. Outro aspecto crucial e que implica restrição é a capacidade das baterias que alimentam o dispositivo, as quais são bastante demandadas pelos circuitos de recepção e transmissão de sinal de rádio – RF – e precisam ser poupadas com a redução do uso do processador e dos acessos à memória.

Em suma, o processo de desenvolvimento de *software* para dispositivos móveis deve levar em consideração todos esses aspectos, uma vez que tornam o ambiente e as informações e sistemas nele residentes sujeitos a vulnerabilidades e representam riscos para a segurança da informação.

Com esse universo de possibilidade de falhas e vulnerabilidades, é imprescindível a automação da validação do software. Um exemplo de boa iniciativa nesse sentido é o “Proyecto Marvin” (fundacionsadosky.org.ar), um software de código aberto para analisar a segurança de aplicativos para o Android.

Tema 5: Internet das Coisas

A Internet das Coisas – IoT – *Internet of Things*, ou a computação ubíqua, é ao mesmo tempo uma tecnologia emergente e algo que já vem sendo incorporado ao nosso cotidiano há décadas, porém de modo discreto, especialmente em função da precária infraestrutura de comunicação de que dispomos. Trata-se da conectividade total, a partir da qual os mais diversos dispositivos de computação – móvel ou não – e eletro/eletrônicos fazem uso de redes – com e sem fio – para produzir, acessar, fornecer e receber informações em tempo real.

Estamos tratando não apenas de eletrodomésticos como geladeiras, lavadoras de roupa, cafeteiras, fogões, televisores, equipamentos de som e vídeo, mas também de lâmpadas, interruptores, alarmes, portas, carros, equipamentos médicos e tudo o mais que possa ser imaginado, por ora bastando que tenha por alimentação a energia elétrica!

Basta essa constatação para imaginar a quantidade de informação – e riscos para a segurança – que essa tecnologia incorpora, e a grande necessidade de prover segurança no software desenvolvido e instalado nesses dispositivos e equipamentos.

Também é notório que os mecanismos e as práticas adotadas para os computadores já não são suficientes para garantir a segurança da informação e dos sistemas nesse novo mundo da IoT. É um novo paradigma para a computação e para os desenvolvedores de software, e novos modelos, métodos, técnicas e ferramentas estão em processo de desenvolvimento ou adaptação para fazer frente a essa demanda.

Trocando Ideias

No fórum “Segurança dos sistemas e do software”, no UNIVIRTUS, troque informações com os colegas com base nos seguintes questionamentos:

1) Existem modelos de desenvolvimento mais seguros que outros?

Por quê?

2) Qual a importância do teste de software para a segurança dos sistemas?

3) A segurança de um sistema permanece inalterada ou no mesmo nível com o passar do tempo? Por quê?

4) Os dispositivos móveis como *smartphones* e *tablets* são mais seguros que os computadores pessoais? Por quê?

5) A Internet das Coisas representa riscos adicionais para a segurança da informação? Por quê?

Na Prática

Para avaliar e aplicar seus conhecimentos, faça um pequeno resumo – uma página de texto, no máximo – dos temas estudados e busque na mídia e na internet casos e exemplos nos quais esses temas são abordados. Apresente suas conclusões para os colegas no fórum “Segurança dos sistemas e do software” e expresse sua opinião sobre o que eles produziram.

Síntese

Nesta aula, foram apresentados os tópicos vinculados à segurança dos sistemas e do software em geral em todo seu ciclo de vida, passando pelo desenvolvimento, operação e manutenção, considerando também a gestão de configuração e de mudança. Além disso, foram abordadas as questões relativas ao software para dispositivos móveis e a Internet das Coisas.

Referências

ABNT. **Segurança da informação – coletânea eletrônica**. Rio de Janeiro: ABNT, 2014.

COSTA, G. C. G. **Negócios eletrônicos**: uma abordagem estratégica e

gerencial. Curitiba: Editora Intersaberes, 2013.

GARTNER GROUP. **Worldwide Smartphones Sales to End Users By OS**. Dezembro, 2014. Disponível em: <http://techcrunch.com/2014/12/15/gartner-301m-smartphones-sold-in-q3-as-xiaomi-muscles-into-the-top-5-at-samsungs-expense/>. Acesso em: 21 maio 2015.

GALVÃO, M. C. **Fundamentos em segurança da informação**. São Paulo: Pearson Education, 2015.

GOERTZEL, K. M.; WINOGRAD, T.; HAMILTON, B. A. **Safety and Security Considerations for Component-Based Engineering of Software-Intensive Systems**. Disponível em: <https://buildsecurityin.us-cert.gov/sites/default/files/NOSSA-SafeSecureSWComposition-02012011.pdf>. Acesso em: 20 maio 2014.

ITU. **ITU Key 2005-2014 ICT data**. Disponível em: <http://www.itu.int/en/ITU-D/Statistics/Documents/statistics/2014/ITU_Key_2005-2014_ICT_data.xls>. Acesso em: 22 maio 2015.

LA POLLA, M.; MARTINELLI, F.; SGANDURRA, D. A Survey on Security for Mobile Devices. **IEEE Communications Surveys & Tutorials**, v.15, n. 1, p. 446-471, 2013.

MUNASSAR, N.; GOVARDHAN, A. A Comparison Between Five Models Of Software Engineering. **International Journal of Computer Science Issues**, v. 7, Issue 5, p. 94-101, set. 2010.

OLIVEIRA, F. B. (Org.). **Tecnologia da informação e da comunicação**: a busca de uma visão ampla e estruturada. São Paulo: Pearson Educational,

2007.

SOMMERVILLE, I. **Engenharia de software**. 6. ed. São Paulo: Pearson-Addison Wesley, 2003.

TAURION, C. **Sua empresa está preparada para o BYOD? IBM Developer Works**. Disponível em: <https://www.ibm.com/developerworks/community/blogs/ctaurion>. Acesso em: 23 maio 2015.