Opinião: 7 características de um aplicativo seguro para smartphones
*Dan Kuykendall**

Os erros mais comuns ocorrem em gerenciamento de sessão. Os erros em si não apresentam um alto risco - mas quanto mais erros, mais vulnerável é a aplicação.

Quando se trata de construir aplicações móveis seguras, os erros mais comuns ocorrem em gerenciamento de sessão. Esses erros em si não apresentam um risco significativo, mas quanto mais erros, mais vulnerável é a aplicação. E é aí que está o problema: muitas vezes eu encontro vários desses erros em qualquer aplicativo fornecido.

Na minha próxima palestra durante a OWASP AppSecUSA 2013, vou dar detalhes de um exemplo de um aplicativo móvel vulnerável. No caso, um popular aplicativo de futebol que, ao ser hackeado, permite que indivíduos mudem as escalações da equipe e postem comentários falsos.

Os usuários que não atualizaram seu aplicativo móvel para a versão mais recente estão em risco de ter seus jogadores manipulados por outros treinadores ou por hackers encrenqueiros.

Claro que isso foi divertido para mim, mas também esclarecedor. Minha investigação continua em curso com o objetivo de desenterrar padrões de vulnerabilidades de gerenciamento de sessão.

Enquanto isso, aqui está uma lista com base em erros comuns que os desenvolvedores e profissionais de segurança podem usar para garantir o gerenciamento adequado de sessões em aplicações móveis.

1. Não confie no cliente.

O servidor deve desconfiar de cada requisição da aplicação, tratando-o como um possível ataque de carga (payload). Assim, o servidor deve confirmar a autenticidade de cada solicitação.

2. Exija criptografia

Para evitar que os atacantes sejam capazes de ler as comunicações wireless a partir de um dispositivo móvel, use SSL para criptografar o cliente e exigir um certificado móvel que pode ser validado.

3. Expirar sessões

Os desenvolvedores geralmente permitem que sessões de aplicativos móveis permaneçam ativas por um tempo muito longo para que os usuários não tenham que logar novamente. No entanto, enquanto a sessão estiver ativa, crackers podem fazer pedidos maliciosos para o servidor. Desenvolvedores estão, em essência, trocando segurança por um pequeno inconveniente para os usuários.

4. Guarde segredo

Um segredo compartilhado conhecido apenas pelo cliente e servidor, e utilizado pelo cliente para assinar as solicitações, impede que o servidor aceite aquelas que foram modificadas.

5. Limite a quantidade de tempo que uma solicitação é válida.

Quanto mais tempo um pedido é válido, maior o risco de um intruso interceptá-lo e modificá-lo. Todos os pedidos devem ser tempo verificados no lado do cliente e expirados após um período de tempo, tal como definido no lado do servidor.

6. Não permitir solicitações repetidas

Atacantes podem reproduzir solicitações interceptadas. O impacto resultante pode variar entre ser meramente uma irritação (como quanto ocorre com um tuite repetido) e ter consequências desastrosas (por exemplo, quando um pedido de transferência de dinheiro é re-enviado).

Os desenvolvedores podem evitar pedidos repetidos usando um NONCE (número usado apenas uma vez). O cliente gera um número aleatório para cada solicitação. O servidor mantém o controle desses números para garantir que os pedidos sejam verdadeiros e não sejam re-executados.

Se ocorrer uma repetição de NONCE, então o servidor sabe que o pedido é inválido. A lista de valores aleatórios armazenados no servidor podem ser minimizados com o uso de um timestamp, como explicado no item 5 dessa lista.

7. Não autorize pedidos modificados

Em vez de repetir um pedido, um atacante pode optar por modificá-lo. Por exemplo, o cracker pode transferir dinheiro para uma conta diferente. Isto pode ser evitado por meio de um segredo compartilhado ou com o uso de chaves de criptografia. Criando um HMAC do pedido e enviando-o ao servidor com o pedido permite que o servidor confirme que o pedido não foi modificado.

Conclusão

Para construir aplicações móveis fortes, os desenvolvedores devem verificar cada medida desta lista.

Qualquer uma que for deixada de lado, sem solução, deixa um aplicativo aberto para ataques ou abusos. No entanto, é fácil ver como todos eles funcionam juntos: por desconfiar do cliente (nº 1) e assumindo que um atacante está espionando as comunicações (nº 2), o desenvolvedor vai querer limitar a quantidade de tempo que um cracker tem para atacar a aplicação (nº 3). A chave secreta (nº 4) pode ser usada para assinar o conteúdo (nº 7), incluindo o timestamp (nº 5) e o NONCE (nº 6) para garantir que nenhum desses três pontos de dados foram modificados.

*Dan Kuykendall é co-CEO e CTO da NT OBJECTives, Inc. Sua palestra, "Revenge of the Geeks: Hacking Fantasy Sports Sites", será um dos destaques do AppSec EUA 2013, que acontece nos dias 18 a 21 de novembro, em Nova York.

Fonte: IDGNOW [Portal]. Disponível em: <http://idgnow.uol.com.br/mobilidade/2013/09/24/opiniao-7-caracteristicas-de-um-aplicativo-seguro-para-smartphones/>. Acesso em: 26 set. 2013.