

Permissões

CONTROLO POR PERMISSÕES

- Mecanismo dos sistemas operativos mobile que permitem algum nível de controlo sobre a atuação das aplicações
- Condiciona o acesso a funcionalidades de hardware (camara, GPS) ou a dados (fotos, informação pessoal) mediante autorização do utilizador
- É um mecanismo útil, mas com limitações
- Problemas e abusos (malware) levou a alterações ao seu funcionamento
 - aceitação de permissões (até ao KitKat) era um bloco
 - versões recentes Android permitem tratamento individual

Android: modo de funcionamento

- Programadores indicam as permissões necessárias ao funcionamento das aplicações
- Lista de permissões incluída no manifesto das aplicações
- Utilizador, ao instalar aplicações, pode consultar as permissões solicitadas
- Utilizador decide se autoriza cada permissão, quando solicitado pela aplicação e de forma individual

Grupo de permissões

Android

- Disponíveis 2 níveis de permissões: normais e perigosas
- Permissões normais englobam atividades que envolvem muito baixo risco para a privacidade dos utilizadores (ex: mudar timezone)
- Permissões perigosas englobam atividades que envolvem dados dos utilizadores ou a capacidade de influenciar o funcionamento de outras aplicações
 - requerem autorização explícita do utilizador
- Autorização pode ser dada
 - no momento de instalação (android 5.1 ou inferior)
 - durante a utilização (android 6 ou superior)
- Mas sempre para todas as permissões do grupo!

Permissões perigosas

(e respetivos grupos)

Grupo	Permissões
CALENDAR	READ_CALENDAR, WRITE_CALENDAR
CAMERA	CAMERA
CONTACTS	READ_CONTACTS, WRITE_CONTACTS, GET_ACCOUNTS
LOCATION	ACCESS_FINE_LOCATION, ACCESS_COARSE_LOCATION
MICROPHONE	RECORD_AUDIO
PHONE	READ_PHONE_STATE, CALL_PHONE, READ_CALL_LOG, WRITE_CALL_LOG, ADD_VOICEMAIL, USE_SIP, PROCESS_OUTGOING_CALLS
SENSORS	BODY_SENSORS
SMS	SEND_SMS, RECEIVE_SMS, READ_SMS, RECEIVE_WAP_PUSH, RECEIVE_MMS
STORAGE	READ_EXTERNAL_STORAGE, WRITE_EXTERNAL_STORAGE

Apps potencialmente perigosas

Aplicações que:

1. Enganam o utilizador para que autorize permissões que não necessitam
2. Escondem comportamento malicioso por detrás de permissões legítimas
3. Tentam levar o utilizador a fornecer informação sensível (ex: cartão de crédito)

Proteção passa por:

- Usar apenas lojas fidedignas (ex: Google, amazon,...)
- Analisar criteriosamente as permissões antes da sua instalação
 - quando em duvida, validar comentários, classificação, página do programador
- Analisar o comportamento da aplicação em tempo de execução

Comunicações

Análise de comportamento

(ponto de vista dos utilizadores)

- Aplicações podem ser analisadas também quanto ao seu comportamento, validando:
 - em que situações solicitam autorização para permissões
 - que tipo de comunicações usam (HTTP, HTTPS, outras)
 - que tipo de API (ou Web APIs) são utilizadas e para que servem
 - que informações recolhem

(ponto de vista das Apps)

- Análise possibilita também a descoberta de problemas na forma como são implementadas
 - uso de APIs sem autenticação
 - Possibilidade de extração de informação de outros utilizadores

(ponto de vista das comunicações)

- Generalidade as Apps
 - usa HTTPS como forma de comunicação segura com os seus serviços
 - solicita autenticação aos utilizadores (muitas vezes OAuth)
- Levanta algumas considerações
 - captura de tráfego de rede para análise não é viável por estar encriptado
 - não é possível utilizar apps e respetivas APIs sem credenciais

Análise de comunicações

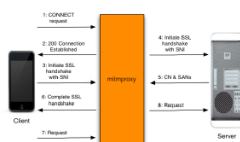
- Captura de tráfego de rede permite analisar
 - pedidos não cifrados
 - equipamentos utilizados
- Ideal é poder-se analisar todos os pedidos efetuados, mesmo os cifrados (HTTPS)
 - solução passa por utilizar software específico para o efeito

A partir do browser

- Análise de comunicações web e browser é simples
- Existem add-ons que possibilitam a análise dos pedidos HTTP enviados/recebidos
- Ferramentas de desenvolvimento dos próprios browsers
- Como executam no browser, a cifra das comunicações não impede o seu uso

mitmproxy

Funcionamento



1. Cliente liga-se ao mitmproxy e faz pedido de ligação
2. mitmproxy responde **200 Ok**, simulando conclusão da ligação
3. Cliente assume estar ligado ao servidor real e inicia ligação segura (TLS) indicando SNI^a
4. mitmproxy cria ligação segura ao servidor real usando o SNI
5. Servidor responde, indicando os campos CN e SAN, utilizados para gerar certificado forjado
6. mitmproxy gera certificado forjado e conclui ligação segura com o cliente (suspensão desde passo 3)
7. Cliente envia o pedido cifrado HTTP para o mitmproxy
8. mitmproxy reenvia pedido para o servidor real