The IoT Hacker’s Handbook

The IoT Hacker’s Handbook

Um guia prático para Hacking da internet das coisas

**Introdução**

Os dez capítulos deste livro abrangem uma série de tópicos, desde exploração de hardware e firmware embarcado até comunicação por rádio, incluindo exploração de BLE e ZigBee.

Para mim, escrever este livro foi uma jornada emocionante e cheia de aventura, compartilhando minhas experiências e as várias coisas que aprendi em minha carreira profissional e colocando tudo nesses dez capítulos.

Espero que você possa aproveitar ao máximo este livro e eu altamente te encorajo a pegar todos os conjuntos de habilidades aprendidos aqui e aplicá-los a problemas do mundo real, ajudando a tornar o ecossistema da Internet das Coisas (IoT) mais seguro. São as contribuições individuais que nos ajudarão a criar um mundo mais seguro, e você, que está lendo este livro, pode fazer parte disso.

Ninguém é perfeito, e este livro certamente terá um ou dois pequenos erros. Se você encontrar algum desses erros, por favor, me avise e ficarei feliz para corrigi-los em futuras edições do Manual do Hacker da IoT.

Eu também dou aulas de três e cinco dias sobre exploração de IoT ofensiva, que eu recomendo que você participe para obter experiência prática com tudo o que é abordado no livro. Para mais informações sobre o treinamento online e as aulas ao vivo, acesse attifystore.com.

A última e mais importante parte é a comunidade! Para você, leitor, eu quero que esteja disposto o suficiente para compartilhar seu conhecimento com seus colegas ou mesmo com alguém que é novo nesta área. É assim que nós, como comunidade, cresceremos.

É tudo da minha parte. Mais uma vez, obrigado por ler o Manual do Hacker da IoT e desejo a todos o melhor em seus empreendimentos de exploração de IoT.

Aditya Gupta (@adi1391)

Fundador e Chefe Hacker,

Attify

**Agradecimentos**

Esse livro nunca teria sido terminado sem o meu time incrível da Attify, que trabalharam dia e noite para ter certeza que produzimos um conteúdo de qualidade como uma equipe.

**1.Internet das coisas: Um primer**

No mundo da tecnologia da comunicação, dois dos eventos que possuem significância especial são a invenção da ARPANET, uma rede de computadores que permite a troca de dados entre computadores mesmo quando estão geograficamente separados, e a ascensão da Internet das Coisas (IoT). Este último, no entanto, foi um processo evolutivo ao invés de um único evento.

As primeiras implementações do conceito de IoT ocorreram quando dois estudantes da Universidade Carnegie Mellon encontraram uma maneira de monitorar o número de latas restantes em uma máquina de venda automática permitido que dispositivos se comuniquem com o mundo externo. Eles fizeram isso adicionando um fotossensor ao dispositivo que contaria toda vez que uma lata saía da máquina, e assim, o número de latas restantes era calculado. Hoje em dia, os dispositivos IoT são capazes de monitorar sua frequência cardíaca e até mesmo controlá-la, se necessário, em caso de um evento adverso. Além disso, alguns dispositivos IoT agora podem servir como fonte de evidência durante julgamentos em tribunais, como visto no final de 2015, quando os dados do FitBit de uma mulher foram usados em um julgamento de assassinato. Outros incidentes incluem o uso de dados de marcapasso e gravações do Amazon Echo em vários julgamentos. A jornada dos dispositivos IoT de um dormitório universitário para dentro de seres humanos é fascinante, para dizer o mínimo. Kevin Aston, quando mencionou o termo Internet das Coisas pela primeira vez, provavelmente não imaginaria que esses dispositivos logo ultrapassariam a população humana total em número. Aston mencionou o termo em referência à tecnologia de identificação por radiofrequência (RFID), que estava sendo usada para conectar dispositivos entre si. A definição de IoT mudou desde então, com diferentes organizações dando seu próprio significado ao termo. A Qualcomm e a Cisco criaram o termo Internet of Everything (IoE), que alguns acreditam ser para uma agenda de marketing. O termo, de acordo com eles, significa estender o conceito de IoT de ser limitado à comunicação máquina-a-máquina para máquinas se comunicando com máquinas e com o mundo físico.

O primeiro vislumbre da IoT atual foi visto em junho de 2000, quando a primeira geladeira conectada à Internet, a Internet Digital DIOS, foi revelada pela LG. A geladeira continha uma tela TFT-LCD de alta qualidade com uma série de funcionalidades, incluindo exibir a temperatura dentro da geladeira, fornecer pontuações de frescura dos itens armazenados e usar a funcionalidade da webcam para rastrear os itens armazenados. O dispositivo inicial que provavelmente chamou mais atenção da mídia e dos consumidores foi o Nest Learning Thermostat em outubro de 2011. Este dispositivo era capaz de aprender a programação do usuário para ajustar diferentes temperaturas desejadas em diferentes horas do dia. A aquisição desta empresa de termostato IoT pelo Google por US$ 3,2 bilhões foi o evento que conscientizou o mundo sobre a revolução tecnológica que se aproximava.

Logo, havia centenas de novas startups tentando interconectar todos os diferentes aspectos do mundo físico a dispositivos e grandes organizações criando equipes internas especializadas para criar sua própria linha de dispositivos IoT para serem lançados no mercado o mais rápido possível. Essa corrida para criar novos dispositivos chamados de inteligentes nos leva ao presente, onde podemos interagir com nossas smart TVs em casa enquanto tomamos uma xícara de café preparada por uma máquina de café controlada pela Internet e controlamos as luzes pela música tocando em seu assistente inteligente. A IoT, no entanto, não se limita apenas ao nosso espaço físico. Ele também possui inúmeras aplicações em empresas, lojas de varejo, saúde, indústria, redes de energia e até mesmo em pesquisas científicas avançadas.

Os formuladores de políticas do mundo digital lutaram contra o ritmo acelerado do surgimento de dispositivos IoT e não conseguiram criar controles de qualidade e regulamentações de segurança rígidos. Isso só mudou recentemente, quando organizações como a GSMA criaram diretrizes de segurança e privacidade para dispositivos IoT, e a Comissão Federal de Comércio (FTC) Foram definidas etapas a serem seguidas para garantir a segurança. No entanto, a demora levou à adoção generalizada de dispositivos de IoT em todos os setores, e também permitiu que os desenvolvedores ignorassem as considerações de segurança no que diz respeito a esses dispositivos. Não foi até o efeito generalizado do botnet Mirai que as deficiências de segurança desses dispositivos se tornaram aparentes Dispositivos seriam vulneráveis. Mirai era uma botnet que atacava dispositivos de Internet das Coisas (IoT), principalmente câmeras conectadas à internet, verificando o status das portas 23 e 2323 e forçando a autenticação por força bruta usando credenciais comuns. Sem surpresa, muitas das câmeras IP expostas à internet tinham acesso telnet disponível com um nome de usuário e senha extremamente comuns, o que era fácil de encontrar. A mesma botnet também foi usada posteriormente para assumir o controle da infraestrutura de internet da Libéria, bem como da DYN, o que levou a um ataque a vários sites populares, incluindo GitHub, Twitter, Reddit e Netflix.

Nos últimos anos, embora a segurança desses dispositivos tenha melhorado lentamente, ainda não atingiu um ponto onde eles possam ser considerados extremamente seguros de usar. Em novembro de 2016, quatro pesquisadores de segurança - Eyal Ronen, Colin O'Flynn, Adi Shamir e Achi-Or Weingarten - criaram um interessante worm de prova de conceito (PoC) que atacava usando drones e tomava o controle das luzes inteligentes Philips Hue de um prédio comercial. Mesmo que o ataque tenha sido apenas um PoC, não é exagero pensar que veríamos ransomware para dispositivos inteligentes semelhante ao WannaCry, pedindo dinheiro para abrir a fechadura da porta ou ligar um marcapasso. Quase todos os dispositivos inteligentes comprovadamente possuem problemas críticos de segurança e privacidade, incluindo sistemas de automação residencial inteligente, dispositivos vestíveis, monitores de bebês e até mesmo brinquedos sexuais pessoais. Considerando a quantidade de dados íntimos que esses dispositivos coletam, é assustador ver o quanto estamos expostos a ataques cibernéticos.

O aumento de incidentes de segurança em dispositivos IoT também levou a uma demanda crescente por profissionais de segurança IoT, atuando tanto na construção quanto na quebra desses sistemas. Isso permite que as organizações garantam que seus dispositivos estejam protegidos das vulnerabilidades que invasores mal-intencionados podem usar para comprometer seus sistemas. Além disso, várias empresas começaram a oferecer recompensas por bugs (bug bounties) para incentivar pesquisadores a avaliar a segurança de seus dispositivos IoT, algumas até mesmo enviando dispositivos físicos gratuitos para esses pesquisadores. Nos próximos anos, essa tendência deve crescer e, com o aumento das soluções de IoT no mercado, haverá uma demanda maior por profissionais especializados em segurança IoT no mercado de trabalho.

**Problemas anteriores de segurança de IoT**

A melhor forma de aprender sobre segurança desses dispositivos é olhar o que aconteceu no passado. Ao aprender sobre os erros de segurança cometidos por outros desenvolvedores de produtos no passado, podemos entender quais tipos de problemas de segurança devemos esperar no produto que estamos avaliando. Mesmo que alguns termos pareçam desconhecidos aqui, nós os discutiremos em detalhes nos próximos capítulos.

**Termostato Nest**

O artigo “Smart Nest Thermostat: Um Espião Inteligente em Sua Casa”, de Grant Hernandez, Orlando Arias, Daniel Buentello e Yier Jin, menciona algumas das deficiências de segurança do Google Nest que permitiam a instalação de um novo firmware malicioso no dispositivo. Isso era feito pressionando o botão do Nest por cerca de 10 segundos para acionar a reinicialização global. Nesse estágio, o dispositivo poderia ser configurado para procurar firmware em mídia USB comunicando-se com o pino sys\_boot5. Na unidade USB, estava presente um firmware malicioso, que o dispositivo então utilizava durante a inicialização.

Jason Doyle identificou outra vulnerabilidade nos produtos Nest que envolvia o envio de um valor personalizado nos detalhes do identificador de conjunto de serviços Wi-Fi (SSID) via Bluetooth para o dispositivo alvo, o que então travava o dispositivo e acabava por reiniciá-lo. Isso também permitiria a um ladrão invadir a casa durante a reinicialização do dispositivo (cerca de 90 segundos) sem ser flagrado pela câmera de segurança Nest.

**Casa inteligente Philips**

Dispositivos domésticos da Philips sofreram de uma série de problemas de segurança em toda a linha de produtos. Isso inclui o popular worm Philips Hue, criado como uma prova de conceito (PoC) pelos pesquisadores de segurança Eyal Ronen, Adi Shamir, Achi-Or Weingarten e Colin O'Flynn. No PoC, eles demonstraram como as chaves de criptografia simétricas embutidas usadas pelos dispositivos Philips poderiam ser exploradas para obter controle sobre os dispositivos alvo através do ZigBee. Também incluía a infecção automática de lâmpadas Philips Hue colocadas próximas umas das outras.

Em agosto de 2013, Nitesh Dhanjani, um pesquisador de segurança, também surgiu com uma nova técnica para causar blecautes permanentes usando uma técnica de ataque por replay para obter controle dos dispositivos Philips Hue. Ele descobriu essa vulnerabilidade depois de perceber que os dispositivos inteligentes Philips Hue consideravam apenas o MD5 do endereço de controle de acesso de mídia (MAC) como o único parâmetro para validar a autenticidade de uma mensagem. Como o invasor pode facilmente aprender o endereço MAC do host legítimo, ele pode criar um pacote malicioso indicando que veio do host genuíno e com os pacotes de dados com o comando para desligar a lâmpada. Fazer isso continuamente permitiria ao invasor causar um blecaute permanente, sem que o usuário tivesse outra opção a não ser substituir a lâmpada.

O Philips Hue (e muitos outros dispositivos inteligentes hoje) usa uma tecnologia chamada ZigBee para trocar dados entre os dispositivos, minimizando o consumo de recursos. O mesmo ataque que era possível no dispositivo usando pacotes Wi-Fi também seria aplicável ao ZigBee. No caso do ZigBee, tudo o que um invasor precisa fazer é simplesmente capturar os pacotes ZigBee para uma solicitação legítima e simplesmente reproduzi-los para realizar a mesma ação em um momento posterior e assumir o controle do dispositivo. Veremos também como capturar e reproduzir pacotes ZigBee no Capítulo 10.

**Lâmpada inteligente Lifx**

Dispositivos de casa inteligente têm sido um dos alvos de pesquisa mais populares entre a comunidade de segurança. Outro exemplo inicial ocorreu quando Alex Chapman, um pesquisador de segurança da empresa Context, descobriu sérias vulnerabilidades de segurança na lâmpada inteligente Lifx, tornando possível para invasores injetarem pacotes maliciosos na rede, obter credenciais WiFi descriptografadas e assumir o controle das lâmpadas inteligentes sem qualquer autenticação.

Os dispositivos, neste caso, estavam se comunicando usando 6LoWPAN, que é outro protocolo de comunicação de rede (assim como ZigBee) construído sobre o 802.15.4. Para farejar os pacotes 6LoWPAN, Chapman usou um Atmel RzRaven flashed com a imagem de firmware Contiki 6LoWPAN, permitindo que ele examinasse o tráfego entre os dispositivos. A maior parte da troca de dados confidenciais ocorrendo nessa rede era criptografada, o que fazia com que o produto parecesse bastante seguro.

Uma das coisas mais importantes durante o teste de penetração de IoT é a capacidade de olhar para o produto como um todo, em vez de apenas olhar para um único componente para identificar os problemas de segurança. Isso significa que para descobrir como os pacotes estão sendo criptografados na comunicação via rádio, a resposta provavelmente está no firmware. Uma das técnicas para obter o binário do firmware de um dispositivo é despejá-lo por meio de técnicas de exploração de hardware, como JTAG, que abordaremos no Capítulo 6. No caso das lâmpadas Lifx, o JTAG deu acesso ao firmware Lifx, que, quando revertido, levou à identificação do tipo de criptografia, que neste caso era o Advanced Encryption Standard (AES), a chave de criptografia, o vetor de inicialização e o modo de bloco usado para criptografia. Como essas informações seriam as mesmas para todas as lâmpadas inteligentes Lifx, um invasor poderia assumir o controle de qualquer lâmpada e invadir o Wi-Fi porque o dispositivo também estava comunicando as credenciais de Wi-Fi pela rede de rádio, que agora poderia ser descriptografada.

**O Hack do “Jeep”**

O Hack do Jeep é provavelmente o hack de IoT mais popular de todos os tempos. Dois pesquisadores de segurança, Dr. Charlie Miller e Chris Valasek, demonstraram em 2015 como poderiam assumir e controlar remotamente um Jeep usando vulnerabilidades no sistema Uconnect da Chrysler, resultando no recall de 1,4 milhão de veículos pela Chrysler.

O hack completo se aproveitou de muitas vulnerabilidades diferentes, incluindo extensos esforços na engenharia reversa de vários binários e protocolos individuais. Uma das primeiras vulnerabilidades que tornou o ataque possível foi o software Uconnect, que permitia a qualquer pessoa se conectar remotamente a ele por meio de uma conexão celular. A porta 6667 estava acessível com autenticação anônima habilitada e foi encontrada executando D-Bus sobre IP, que é usado para comunicação entre processos. Depois de interagir com o D-Bus e obter uma lista de serviços disponíveis, um dos serviços com o nome NavTrailService tinha um método de execução que permitia aos pesquisadores executar código arbitrário no dispositivo. A Figura 1-1 mostra o código de exploração usado para abrir um shell root remoto na unidade principal.

<http://illmatics.com/Remote%20Car%20Hacking.pdf>

Uma vez que a execução arbitrária de comandos foi obtida, tornou-se possível realizar um movimento lateral e enviar mensagens CAN assumindo o controle dos vários elementos do veículo, como volante, freios, faróis e assim por diante.

**Belkin Wemo**

Belkin Wemo é uma linha de produtos que oferece automação residencial completa aos consumidores.

Belkin Wemo é um caso interessante em que os desenvolvedores tomaram precauções para impedir que invasores instalassem firmware malicioso no dispositivo. As atualizações de firmware para Belkin Wemo, no entanto, aconteciam por um canal não criptografado, o que permitia aos invasores modificar o pacote binário do firmware durante a atualização. Como medida de proteção, o Belkin Wemo utilizava um mecanismo de distribuição de firmware criptografado baseado em GNU Privacy Guard (GPG) para que o dispositivo não aceitasse pacotes de firmware malicioso injetados por um invasor. Essa proteção de segurança foi superada com extrema facilidade porque o dispositivo estava distribuindo a chave de assinatura do firmware junto com o firmware durante o processo de atualização, tudo em um canal não criptografado. Um invasor poderia, portanto, modificar facilmente o pacote, bem como assiná-lo com a chave de assinatura correta, e o dispositivo aceitaria alegremente este firmware. Essa vulnerabilidade foi descoberta por Mike Davis da IOActive no início de 2014 e recebeu uma pontuação (CVSS) de 10.0 pela criticidade da vulnerabilidade.

Mais tarde, descobriu-se que o Belkin Wemo tinha vários outros problemas de segurança, incluindo bugs como injeção de SQL e modificação do nome do dispositivo para executar JavaScript arbitrário no smartphone Android do usuário, entre outros. Pesquisas adicionais foram realizadas no Belkin Wemo pelo grupo FireEye (consulte <https://www.fireeye.com/blog/threatresearch/2016/08/embedded_hardwareha.html>), que envolveu a obtenção de acesso ao firmware e console de depuração usando técnicas de hardware Universal Asynchronous Receiver Transmitter (UART) e Serial Peripheral Interface (SPI). Isso também os levou a identificar que, por meio do acesso ao hardware, alguém pode modificar facilmente os argumentos do bootloader, tornando inútil a verificação de assinatura do firmware do dispositivo.

**Bomba de insulina**

Um pesquisador de segurança chamado Jay Radcliffe, trabalhando para a Rapid7, identificou que alguns dispositivos médicos, especificamente bombas de insulina, poderiam estar sofrendo de uma vulnerabilidade a ataques baseados em replay. Radcliffe, ele próprio diabético tipo 1, decidiu pesquisar uma das bombas de insulina mais populares do mercado, o sistema de bomba de insulina OneTouch Ping da Animas, subsidiária da Johnson & Johnson. Durante a análise, ele descobriu que a bomba de insulina usava mensagens de texto claro para se comunicar, o que tornava extremamente simples para qualquer pessoa capturar a comunicação, modificar a dose de insulina a ser administrada e retransmitir o pacote. Quando ele testou o ataque na bomba de insulina OneTouch Ping, funcionou perfeitamente, sem nenhuma maneira de saber a quantidade de insulina que estava sendo administrada durante o ataque.

A vulnerabilidade foi corrigida pelo fornecedor, Animas, em cinco meses, o que mostra que pelo menos algumas empresas levam os relatórios de segurança a sério e tomam medidas para manter os clientes seguros.

**Fechaduras Inteligentes**

Um pesquisador de segurança com o apelido Jmaxx embarcou em um desafio para encontrar pontos fracos de segurança na fechadura inteligente August, considerada uma das fechaduras inteligentes mais populares e seguras, usada tanto por consumidores em suas casas quanto por anfitriões do Airbnb para permitir que os hóspedes façam o check-in quando for conveniente.

Algumas das vulnerabilidades que ele descobriu incluíam a capacidade dos hóspedes de se tornarem administradores modificando um valor no tráfego de rede de usuário para superusuário, firmware não assinado, funcionalidade do aplicativo para ignorar fixação de Secure Sockets Layer (SSL) (habilitando modo de debug) e muito mais.

No mesmo evento, os pesquisadores de segurança Anthony Rose e Ben Ramsey da empresa de segurança Merculite fizeram outra apresentação intitulada “Arrombando fechaduras Bluetooth Low Energy a 400 metros de distância”, na qual revelaram vulnerabilidades em uma série de produtos de fechaduras inteligentes, incluindo Quicklock Padlock, iBluLock Padlock, Plantraco Phantomlock, Ceomate Bluetooth Smart Doorlock, Elecycle EL797 e EL797G Smart Padlock, Vians Bluetooth Smart DoorlockOkidokey Smart Doorlock, Poly-Control Danalock Doorlock, Mesh Motion Bitlock Padlock e Lagute Sciener Smart Doorlock.

As vulnerabilidades descobertas por Rose e Ramsey eram de vários tipos, incluindo transmissão da senha em texto puro, suscetibilidade a ataques baseados em replay, engenharia reversa de aplicativos móveis para identificar informações confidenciais, fuzzing e spoofing de dispositivo. Por exemplo, durante o processo de redefinição de senha, o Quicklock Padlock envia um pacote Bluetooth Low Energy (BLE) contendo o opcode, a senha antiga e a nova senha. Como até mesmo a autenticação normal, acontece através de comunicação de texto puro, um invasor pode então usar a senha para definir uma nova senha para a fechadura da porta, tornando o dispositivo inútil para o proprietário original. A única maneira de reiniciá-lo seria remover a bateria do dispositivo após abrir o compartimento. Em outro dispositivo, o Danalock Doorlock, é possível fazer engenharia reversa do aplicativo móvel para identificar o método de criptografia e encontrar a chave de criptografia codificada ("thisisthesecret") usada.

**Hackeando armas e rifles inteligentes**

Além dos dispositivos e eletrodomésticos inteligentes típicos, os rifles também estão ficando inteligentes. A TrackingPoint, fabricante de tecnologia para rifles inteligentes, oferece um aplicativo móvel para visualizar e ajustar a mira do tiro. Este aplicativo foi considerado vulnerável a alguns problemas de segurança. Runa Sandvik e Michael Auger identificaram vulnerabilidades no rifle inteligente que lhes permitiram acessar interfaces de programação de aplicativos de administração (APIs) após obter acesso ao dispositivo via UART. Explorando o aplicativo móvel, um ataque baseado em rede permitiria a um invasor alterar vários parâmetros, como velocidade e direção do vento, peso da bala e outros parâmetros necessários para o disparo. Quando esses parâmetros são modificados, o atirador não saberia que essas alterações foram feitas.

Outro caso ocorreu quando um pesquisador de segurança conhecido como Plore conseguiu burlar algumas das restrições de segurança aplicadas pela IP1, uma arma inteligente da Armatix. A arma inteligente exigia que o atirador usasse um relógio especial fornecido pela IP1 para disparar a arma. Para contornar as restrições de segurança, Plore inicialmente realizou uma análise de sinal de rádio e encontrou a frequência exata que a arma usa para se comunicar. Mais tarde, ele percebeu que usando alguns ímãs, o pino de metal que trava o pino de disparo poderia ser manipulado, permitindo ao atirador disparar a bala. Mesmo que o uso de ímãs não seja um ataque de alta tecnologia que você possa pensar ser necessário para explorar dispositivos IoT, é um ótimo exemplo de como pensar fora da caixa pode ajudar a identificar vulnerabilidades.

Estas vulnerabilidades servem como exemplos para ajudá-lo a entender vários tipos de vulnerabilidades normalmente encontradas em dispositivos IoT. Mais tarde, abordaremos vários componentes de dispositivos IoT, incluindo técnicas para exploração de hardware, rádio, firmware e software. Você aprenderá mais sobre como usar algumas dessas técnicas nos dispositivos IoT que você está pesquisando ou realizando um teste de penetração.

**Fragmentação da Internet das coisas**

Como a IoT é um campo enorme, com toda empresa querendo sua fatia do bolo, você frequentemente encontrará vários protocolos e frameworks que podem ajudar os desenvolvedores a levar seus produtos ao mercado mais rapidamente.

Frameworks de IoT são várias ofertas disponíveis que ajudam os desenvolvedores de IoT a acelerar o processo de desenvolvimento de uma solução de dispositivo IoT, aproveitando a base de código e bibliotecas existentes oferecidas, reduzindo o tempo de lançamento do produto no mercado. Embora isso torne as coisas significativamente mais fáceis para desenvolvedores e empresas, o outro lado, que muitas vezes é negligenciado, é a segurança desses frameworks. Na verdade, com base em minhas experiências com testes de penetração em dispositivos IoT, dispositivos usando vários frameworks eram frequentemente vulneráveis ​​a problemas de segurança básicos. As discussões que tive posteriormente com as equipes de produto revelaram que a mentalidade geral é que, se alguém está usando um framework popular, ele geralmente é considerado seguro por design, resultando em descuido na avaliação de sua segurança.

Não importa de qual lado você esteja, os construtores ou os invasores, é importante observar os problemas de segurança do produto, independentemente da estrutura subjacente ou dos conjuntos de protocolos usados. Por exemplo, você frequentemente encontrará desenvolvedores usando ZigBee pensando que ele é extremamente seguro, deixando seus produtos vulneráveis ​​a todos os tipos de ataques baseados em rádio.

Neste livro, não nos concentramos necessariamente em nenhuma framework ou pilha de tecnologia específica, mas sim em uma abordagem aplicável a qualquer solução de dispositivo IoT, independentemente da arquitetura subjacente. Nesse processo, no entanto, também abordamos alguns protocolos populares (por exemplo, ZigBee e BLE) para lhe dar uma ideia de que tipo de vulnerabilidades esperar e como encontrar esses problemas de segurança.

Algumas das estruturas de IoT mais populares incluem o seguinte:

Eclipse Kura ( https://www.eclipse.org/kura/ )

The Physical Web ( https://google.github.io/physicalweb/ )

IBM Bluemix (now IBM Cloud: https://www.ibm.com/cloud/ )

Lelylan ( http://www.lelylan.com/ )

Thing Speak ( https://thingspeak.com/ )

Bug Labs ( https://buglabs.net/ )

The thing system ( http://thethingsystem.com/ )

Open Remote ( http://www.openremote.com/ )

OpenHAB ( https://www.openhab.org/ )

Eclipse IoT ( https://iot.eclipse.org/ )

Node-Red ( https://nodered.org/ )

Flogo ( https://www.flogo.io/ )

Kaa IoT ( https://www.kaaproject.org/ )

Macchina.io ( https://macchina.io/ )

Zetta ( http://www.zettajs.org/ )

GE Predix ( https://www.ge.com/digital/predixplatform-foundation-digital-industrialapplications )

DeviceHive ( https://devicehive.com/ )

Distributed Services Architecture ( http://iot-dsa.org/ )

Open Connectivity Foundation ( https://openconnectivity.org/ )

Isso é apenas uma pequena fração de alguns dos frameworks de dispositivos IoT mais populares que você encontrará ao mergulhar no mundo da IoT.

Da mesma forma, quando se trata de protocolos de comunicação, há toda uma gama de protocolos sendo usados pelos fabricantes para suas soluções de IoT.

Alguns dos protocolos de comunicação mais populares incluem o seguinte:

* Wi-Fi
* BLE
* Cellular/Long Term Evaluation (LTE)
* ZigBee
* ZWave
* 6LoWPAN
* LoRA
* CoAP
* SigFox
* Neul
* MQTT
* AMQP
* Thread
* LoRaWAN

Para avaliar adequadamente a segurança IoT de um determinado dispositivo ou protocolo de comunicação, você precisará de várias ferramentas de hardware. Por exemplo, o Ubertooth One seria necessário para capturar e analisar pacotes BLE, o Atmel RzRaven para ZigBee, e assim por diante.

Agora que temos uma boa ideia do que é IoT e das várias tecnologias envolvidas, vamos dar uma olhada em alguns dos fatores que levam à insegurança desses dispositivos.

**Razões para Vulnerabilidades de Segurança IoT**

Dado que os dispositivos IoT são extremamente complexos por natureza, é altamente provável que a maioria dos dispositivos que você encontrar terá problemas de segurança. Se tentarmos entender por que essas vulnerabilidades existem em primeiro lugar, e como você pode evitar esses problemas de segurança ao construir um produto, precisamos nos aprofundar em todo o ciclo de vida de desenvolvimento do produto, desde a fase de ideação até o produto ser lançado no mercado.

Alguns dos motivos que se destacam como causa de problemas de segurança ao construir esses dispositivos são apresentados a seguir.

**Falta de Conscientização de Segurança entre Desenvolvedores**

Desenvolvedores que trabalham nestes dispositivos inteligentes frequentemente possuem menos conhecimento, ou sequer conhecimento, sobre as possíveis vulnerabilidades de segurança em dispositivos IoT. Considerando que, em grandes organizações, os desenvolvedores geralmente já estão sobrecarregados, seria uma ótima ideia ter reuniões periódicas para discutir como eles podem construir produtos seguros desde o início, incluindo táticas práticas como diretrizes rígidas de codificação a serem seguidas e uma lista de verificação de segurança para qualquer amostra de código em que trabalhem.

**Falta de Perspectiva Macro**

Como veremos no próximo capítulo sobre os vários componentes que constituem um dispositivo de IoT, é extremamente fácil para desenvolvedores ou equipes de segurança esquecerem o fato de que é a interconexão de dispositivos e diversas tecnologias que pode levar a problemas de segurança. Por exemplo, apenas olhar para o aplicativo móvel pode não revelar problemas de segurança, mas se você combinar as descobertas do aplicativo móvel e como a comunicação de rede funciona, poderá descobrir uma falha crítica de segurança. É essencial que as equipes de produto invistam mais tempo e esforço analisando toda a arquitetura do dispositivo e realizando modelagem de ameaças.

**Problemas de segurança baseados na cadeia de suprimentos**

Uma das causas das vulnerabilidades de segurança em dispositivos IoT é o envolvimento de muitos participantes. Isso significa que você frequentemente encontrará diferentes componentes de dispositivos sendo fabricados por fornecedores diferentes, tudo sendo montado por outro fornecedor e, finalmente, sendo distribuído por outro ainda. Isso, embora inevitável na maioria das situações, pode levar a problemas de segurança (ou backdoor) que podem ser introduzidos por um deles, colocando todo o produto em risco.

**Utilização de Frameworks e Bibliotecas de Terceiros Inseguras**

No caso de dispositivos IoT ou qualquer outra tecnologia, é comum encontrar desenvolvedores usando bibliotecas e pacotes existentes, introduzindo amostras de código potencialmente vulneráveis em um produto seguro. Embora algumas organizações tenham verificações de qualidade no código escrito pelos desenvolvedores, muitas vezes elas tendem a negligenciar os pacotes que estão sendo utilizados. Isso também é acompanhado pelas exigências de negócios de uma organização, onde a gerência exige que os produtos cheguem ao mercado em prazos acelerados (geralmente irreais), o que coloca a avaliação de segurança do produto em segundo plano. Muitas vezes, sua importância não é percebida até que o produto sofra uma violação de segurança.

**Conclusão**

Neste capítulo, vimos o que são dispositivos IoT, os protocolos e frameworks usados por esses dispositivos inteligentes e os motivos pelos quais esses dispositivos geralmente são vulneráveis. Também examinamos alguns dos problemas de segurança identificados anteriormente em soluções populares de dispositivos IoT para entender quais são algumas das vulnerabilidades encontradas em dispositivos do mundo real. No próximo capítulo, examinaremos mais profundamente o mapeamento da superfície de ataque desses dispositivos e como podemos identificar e possivelmente evitar riscos de segurança em dispositivos IoT.