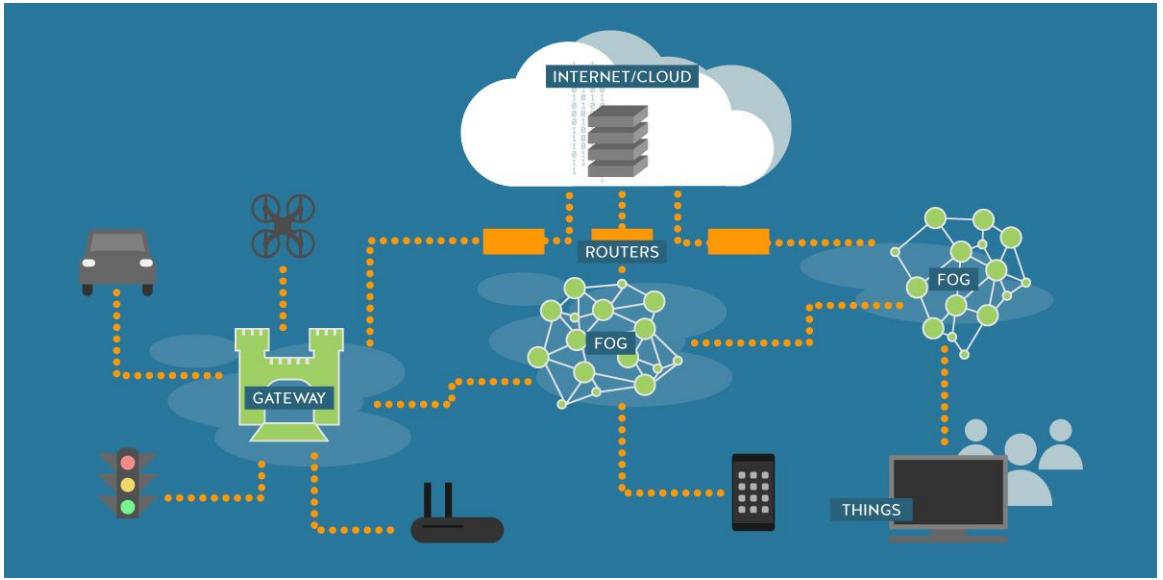
Fog Computing for the Internet of Things: Security and Privacy Issues

Authored By: Arwa Alrawais, Abdulrahman Alhothaily, Chunqiang Hu, and Xiuzhen Cheng

Presented By: Tuhina

Background I

Fog Computing: refers to extending cloud computing to the edge of an enterprise's network. AKA fogging, fog computing facilitates the operation of compute, storage and networking services between end devices and cloud computing data centers.



Background II

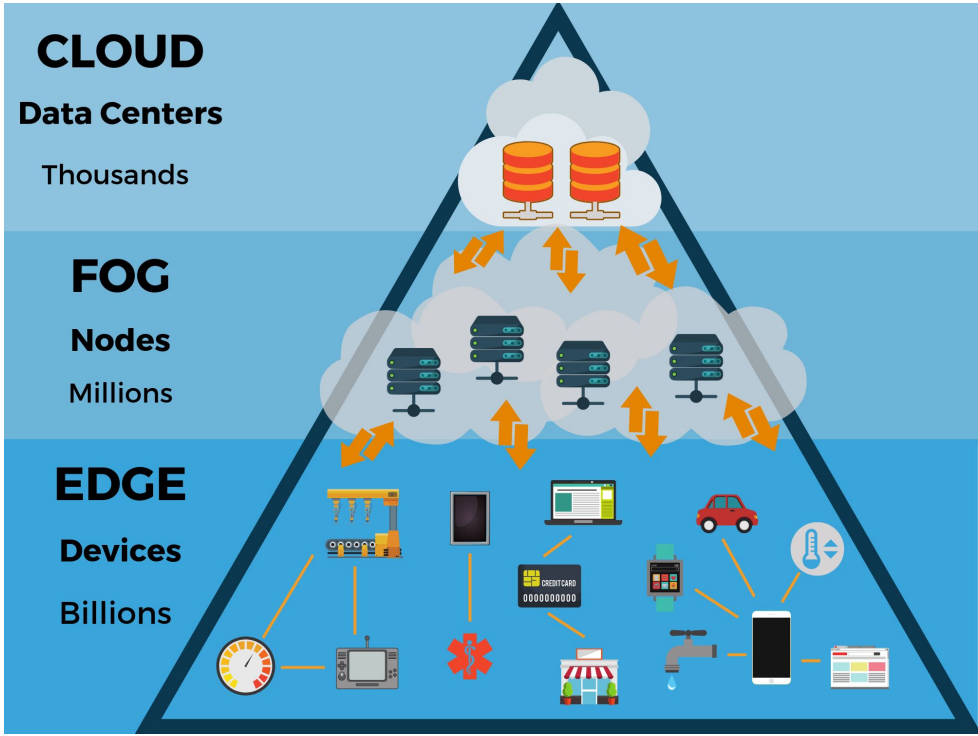
Fog Computing v Edge Computing

Edge computing is a component, or a subset of fog computing.

Fog computing: the way data is processed from where it is created to where it will be stored.

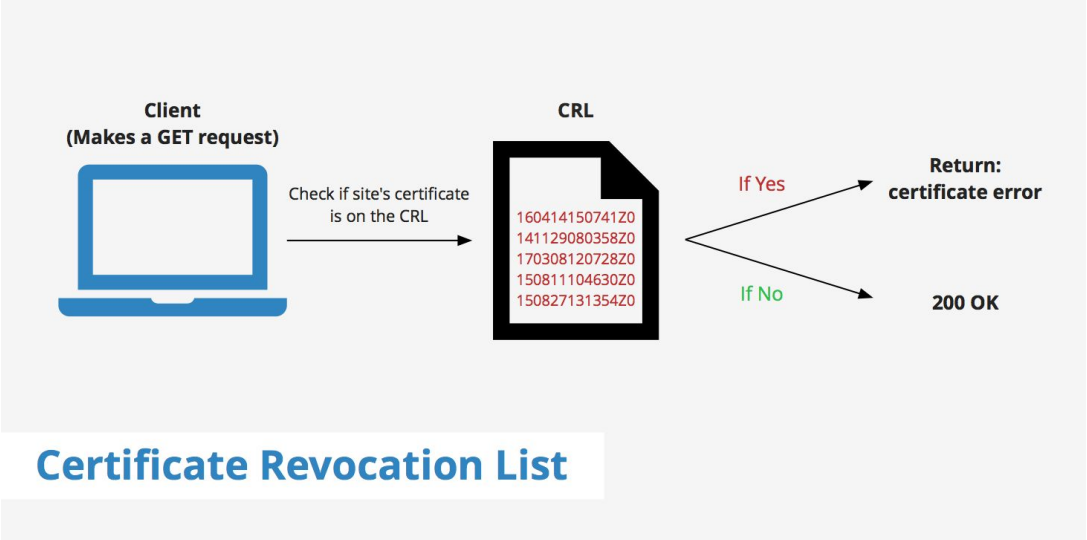
Edge computing: data being processed close to where it is created

Fog computing encapsulates not just that edge processing, but also the network connections needed to bring that data from the edge to its end point.



Background III

Certificate Revocation: is a process of invalidating an issued SSL certificate. Ideally, browsers and other clients should be able to detect that the certificate is revoked in timely manner, show the security warning, that certificate is no longer trusted, and prevent user from further consuming such a website



Why Fog Devices?

Intent: extend the cloud to the network edge and provide efficient data access, computation, networking, and storage; enables a new breed of services at the edge to deliver a wide variety of applications for IoT devices.

Features: supports mobility, location awareness, heterogeneity, large scalability, low latency, improved security, and geodistribution.

Goals: to reduce the data volume and traffic to cloud servers, decrease latency, and improve quality of service (QoS).

Security & Privacy Challenges in IoT I

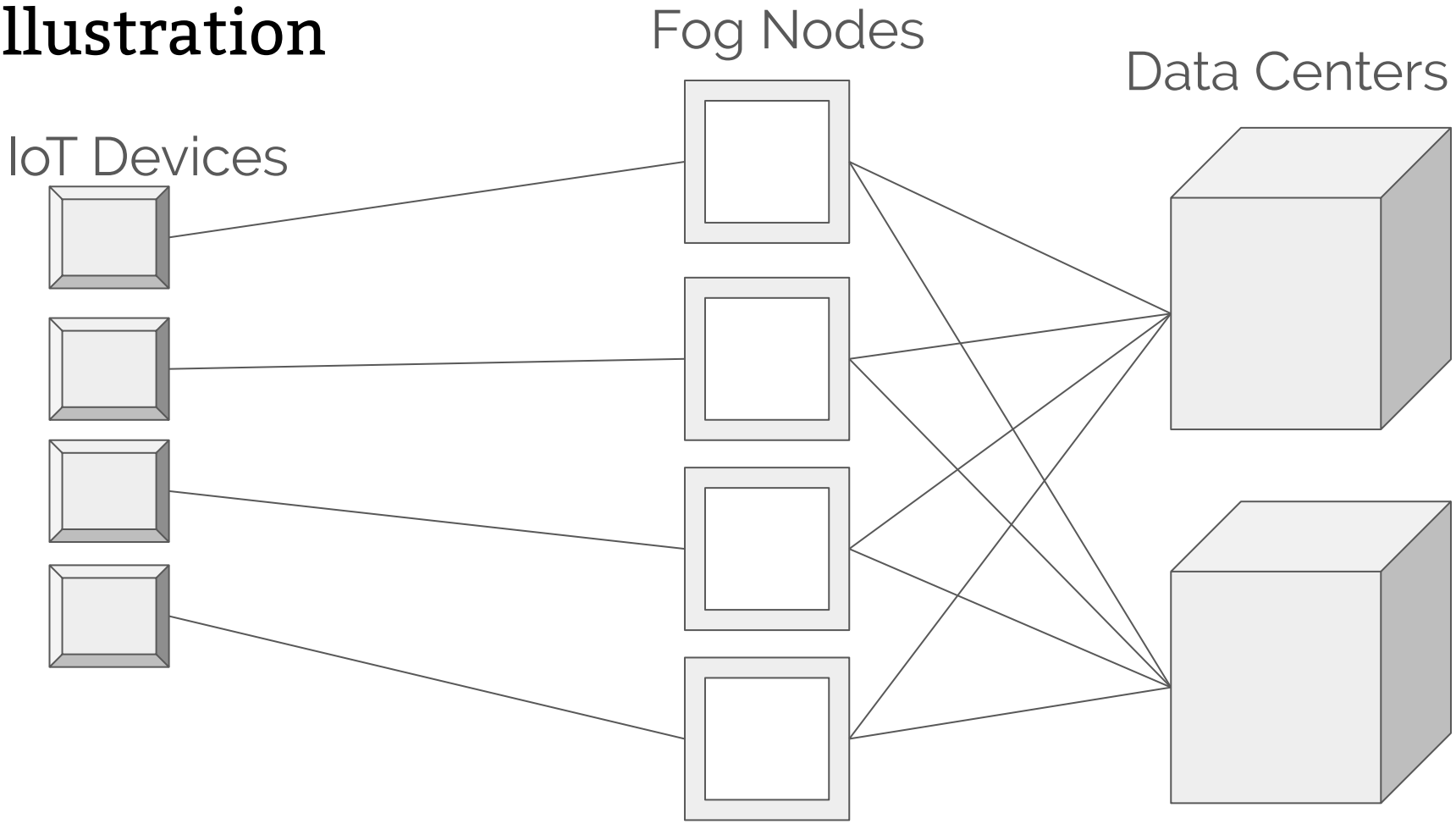
Authentication: IoT devices don't have enough memory and CPU power to execute the cryptographic operations required for an authentication protocol -> can outsource expensive computations and storage to a fog device

Trust: Cultivating the trust between IoT devices plays a central role in establishing secure environments to preserve the security and reliability of IoT services.

Rogue Node Detection: A malicious IoT node could pretend to be legitimate to exchange and collect the data generated by other IoT devices for malicious purposes.

Data Protection: Data must be preserved not only at the communication level, but also at the processing level; data usually is sent to the cloud for further processing and analyzing. The lack capability of IoT devices to encrypt or decrypt makes computing the authenticity and integrity of the data a critical challenge.

Illustration



Security & Privacy Challenges in IoT II

Privacy: IoT devices lack the ability to encrypt or decrypt generated data, which makes it vulnerable to an adversary; the location privacy that can be used to infer the IoT device's location.

Access control: A technique to ensure that only authorized entities can access a certain resource, such as an IoT device, or the collected data.We need access control to make sure that only trusted parties can perform a given action such as accessing IoT device data, issuing a command to an IoT device, or updating IoT device software. **Intrusion detection:** Techniques detect misbehavior or malicious IoT devices and notify others in the network to take appropriate actions. Most of the existing techniques in the IoT target a few attacks with low efficiency.

Problem Definition

Problem: ~2.6 million IoT devices will be used in 2016

- More IoT devices -> more data generated he increasing number of IoT devices will accordingly inflate the amount of data generated.
- IoT devices have low computation power, bandwidth, battery, and storage -> characteristics affect the user experience and QoS.

real-time requirement that can't be fulfilled with traditional cloud computing

Solution: Fog computing (!)

CRL & OCSP

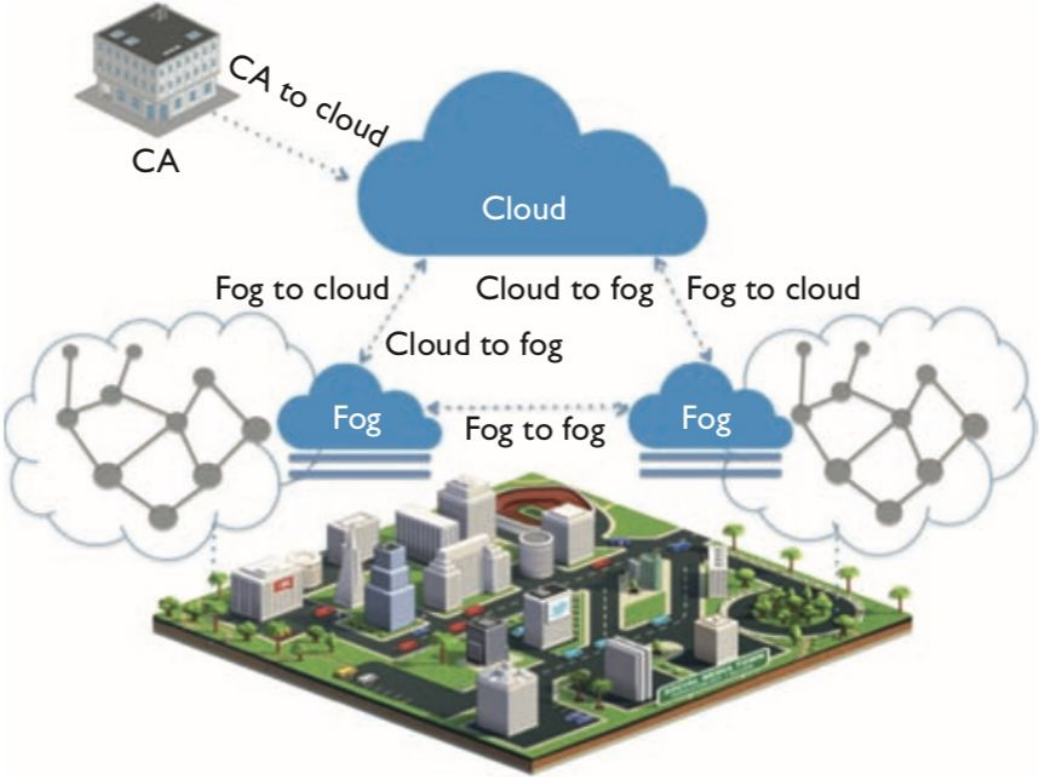
CRL(Certificate Revocation List):

Issues: A CRL file imposes a high burden of communication overhead and requires large storage on the client side. The size of a CRL file correlates with the number of the revoked certificates, and it grows over time. The CRL files' size varies between 793 bytes and 5 Mbytes, and the updated period time is ranging from a few days to a year. Accordingly, a CRL wouldn't perform well in IoT environments due to the resource limitations of IoT devices.

OCSP(Online Certificate Status Protocol):

Issues: OCSP is a real-time protocol, making it more suitable for IoT deployment. However, OCSP requires a round-trip time to check each certificate's status, and IoT devices can't afford to communicate each time with the OCSP server due to resource constraints.

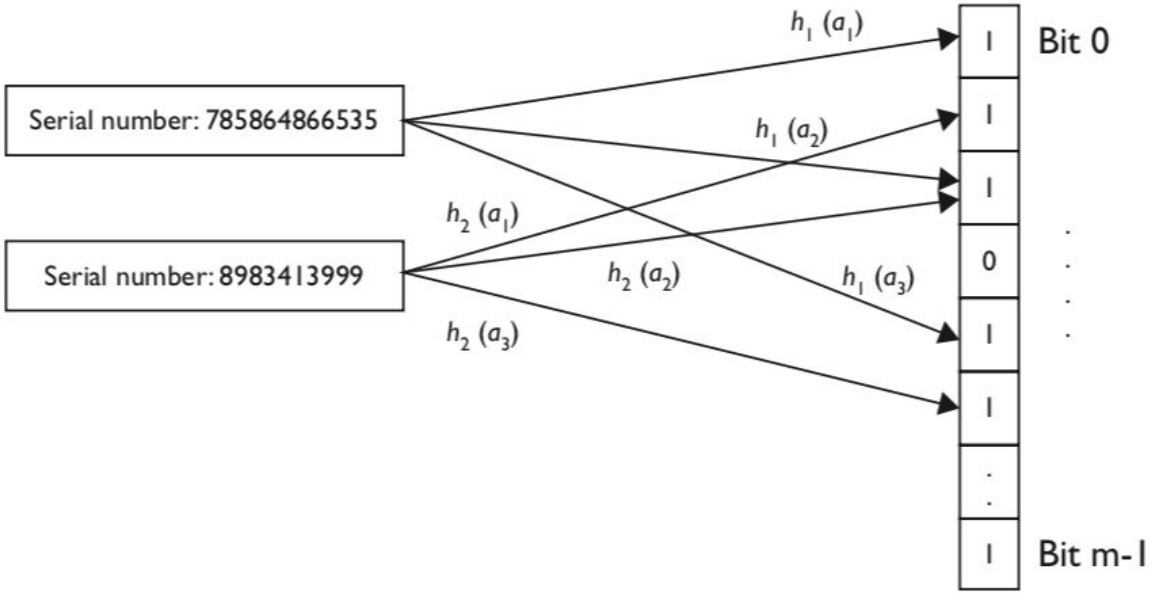
System Design - Certificate Revocation Scheme



each fog node maintains CRL files for a CA -> the node passes the certificates through k hashing functions and store a single bit for each certificate in a bloom filter.

Note: Assume that each fog node is responsible for serving a group of IoT devices that use digital certificates issued by a particular CA

System Design - Bloom Filter



Example of a bloom filter with k = 3 independent hash functions. Although a false positive matching is possible in the bloom filter, a false negative is not.

Explanation - Bloom Filter I

Definition: A bloom filter is a space-efficient data structure to store a group of elements in a bit-vector (0, ..., *m* – 1) that can be used to check whether an element is a member of the group.

"Probabilistic Data Structure" - give you memory-efficient, faster result with a cost of providing a 'probable' result instead of a 'certain' one.

> The bloom filter essentially consists of a bit-vector or bit-list *(a list containing only either 0 or 1-bit value)* of length **m**, initially all values set to 0, as shown below.

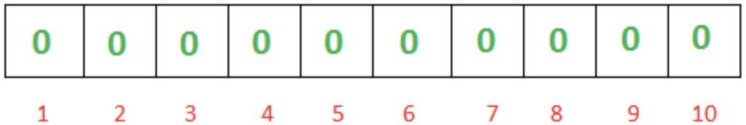


Image Credit: GeeksforGeeks

Applications: bioinformatics, databases, spell-checker

Explanation - Bloom Filter II

Scheme: Use a bloom filter to create a short list that can effectively reduce the revocation list size with acceptable overhead. An empty bloom filter vector is set to 0. To store certificate revocation information, use the certificate's serial number, unique identifier within a CA. The serial number should be hashed with *k* independent hash functions mapping it to a group of bit locations that should be set to 1. False positive matching is possible, a false negative isn't. (Issue)

Flow: CA (creates & signs CRL) ->send-> cloud (manage received revocation lists from multiple CAs) ->forward-> intended fog nodes (each fog node serves a group of IoT devices that hold certificates belonging to a specific CA) stores list & prepares bloom filter that maps the revocation information

Explanation - Bloom Filter III

Problem: false positive matching

Solution: When an IoT device communicates with another device, it needs to verify the device's certificate status. If the certificate's identity isn't in the bloom filter, it means the certificate isn't revoked. However, if the identity is found in the bloom filter, there are two possible cases:

1) the certificate is revoked

2) the certificate isn't revoked and the bloom filter triggers a false positive result.

The fog node acts as a gateway for each IoT group to check the certificate's status. If the certificate is found in the bloom filter, the device needs to contact the fog node to verify the certificate's status. The IoT device sends to the fog a packet containing the intended device's certificate serial number. Then, the fog checks the certificate's serial number against the stored list and replies with the certificate's status.

Evaluation I

Storage: size of the bloom filter is calculated to not have any storage overhead

Privacy: identity obfuscation or designing an efficient privacy-preserving technique based on partitioning the data among fog devices

$$m = \frac{-b \cdot ln(p)}{(ln(2)^2)}$$

where m is the number of bits needed in the bloom filter, and p is the chosen probability of a false positive, which is 0.01 in our experiment.[13]

Evaluation II

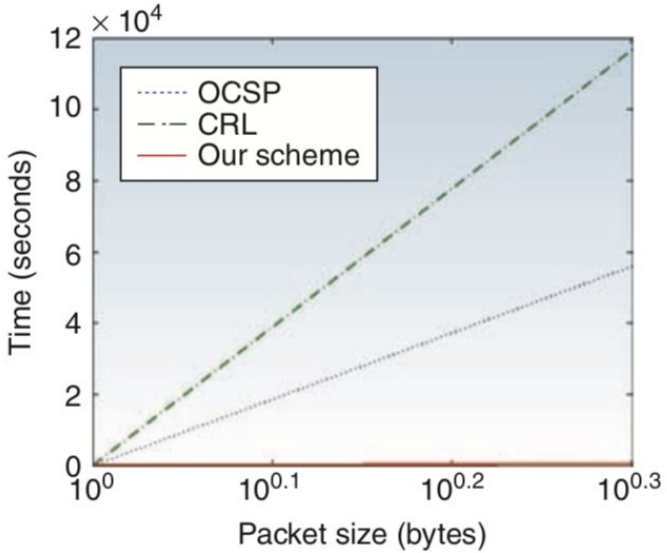


Figure 4. Comparison of the communication overhead in different certificate revocation schemes. Our scheme consumes less bandwidth, which is compatible with the packet size.

Communication Overhead: main contributor is packet size -> decreased

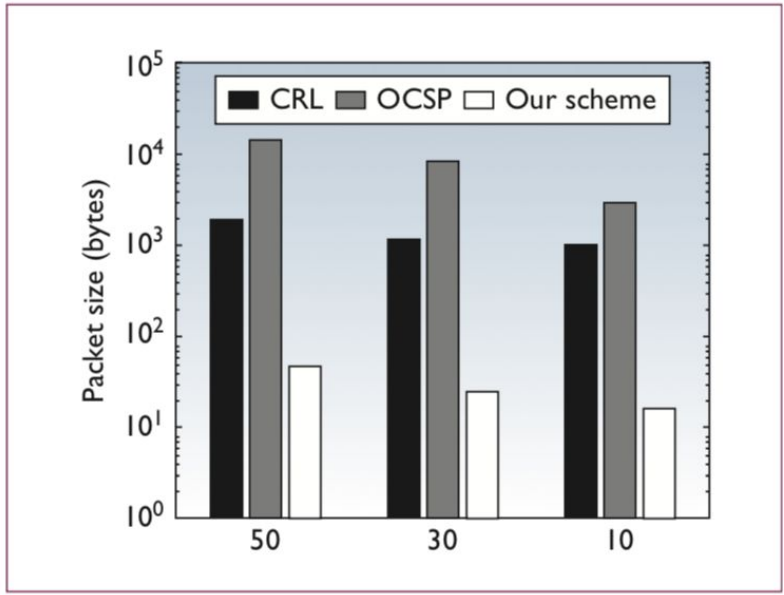


Figure 3. Comparison of the packet size in different certificate revocation schemes when b = 50, b = 30, and b = 10. CRL = certificate revocation list and OCSP = Online Certificate Status Protocol.

Critique

Assumptions:

-in the scheme, assumptions are made about the fog nodes & the quantity of valid certificates they contain (security?)

The paper did not provide critical analysis of fog devices, though they did provide analysis of their proposed fog device schema

The paper mentioned several security concerns w/IoT and then proposed a solution wo/addressing any of them

Proposed Areas of Research I

Privacy: Fog computing could help preserve privacy in the IoT and protect users by minimizing the need to transmit sensitive data to the cloud for analysis; this new model would help in processing the data at the network edge, close to the IoT devices that act on that data. However, this introduces challenges concerning data, location, and user privacy.

Updating IoT devices: Many IoT devices are still vulnerable, and remote software update capabilities need to be designed to handle security updates. Fog computing could be a crucial part of a solution that identifies vulnerabilities and tracks firmware updates in IoT devices. The geodistribution characteristic of fog computing could help supply IoT devices with the necessary security updates to keep them secure.

Secure & Efficient Protocols: Wireless transmissions and security computations consume a significant fraction of the energy budget. The primary challenge is how to design efficient secure schemes in IoT without sacrificing the performance and consuming a high energy.

Proposed Areas of Research II

Authentication: in IoT has several challenges such as scalability and efficiency. Traditional authentication is inefficient, and there's a need for a secure, scalable, efficient, and user-friendly solution to cope with resource-constrained IoT devices.

Attack Detection: Since the fog is an extension of the cloud at the network edge, it's possible to reuse developed detection systems of the cloud in the fog platform. Fog computing provides a new opportunity to design an efficient intrusion-detection solution on both the cloud and IoT device sides.

Location Verification: IoT devices can move in a fast and dynamic way, which complicates location verification. The main challenge is how to design a secure and precise location-verification scheme in harsh environments, and at the same time, the scheme should be suitable for resource constrained IoT devices.

Access Control: Fog would facilitate the adoption of many standard access control models, like an access control list or attribute-based access control in IoT environments. In a distributed architecture, fog computing as an ideal candidate to grant access tokens to authorized parties who use them to perform a given action.

Conclusions I - S&P

The paper defines the following issues in security & privacy (non-exhaustive):

Authentication

Trust

Rogue Node Detection

Data Protection

Privacy

Access Control

Intrusion Detection

Conclusions II - Scheme

Provides a novel method of certificate revocation using fog computing:

The method is each fog node maintains CRL files for a CA -> the node passes the certificates through k hashing functions and store a single bit for each certificate in a bloom filter. The bloom filter data structures are stored in the IoT devices. If a source device talks to a destination device then the bits corresponding to the destination are checked. If they are unset then the destination has a valid certificate; otherwise the fog is checked with because there is a chance for false positives.

Questions ??

Github:

Although the table isn't being stored on the IoT device, is this proposed method scalable? Would the addition of many more devices on the system increase latency as more requests are being made to the single Fog Node? -gkahl

What sort of security problems are exposed by trusted an external device to do crypto on your behalf? What sort of controls can be put in place to defend this, and do these security controls scale down to IOT? -bushidocodes

I understand that fog computing adopts some of the similar challenges faced with cloud computing with respect to IoT devices, as discussed in this paper, but what exactly are the unique challenges that can be introduced that are specific to fog in IoT that is not faced with the cloud? -rachellkm

Regarding their certificate revocation proposal, how would the case of the fog node not being able to reach the cloud be handled? -pcodes