Ben Walker
12-16-18
CSC 495: Reading Question 3 – Blockchain Securing IoT Devices

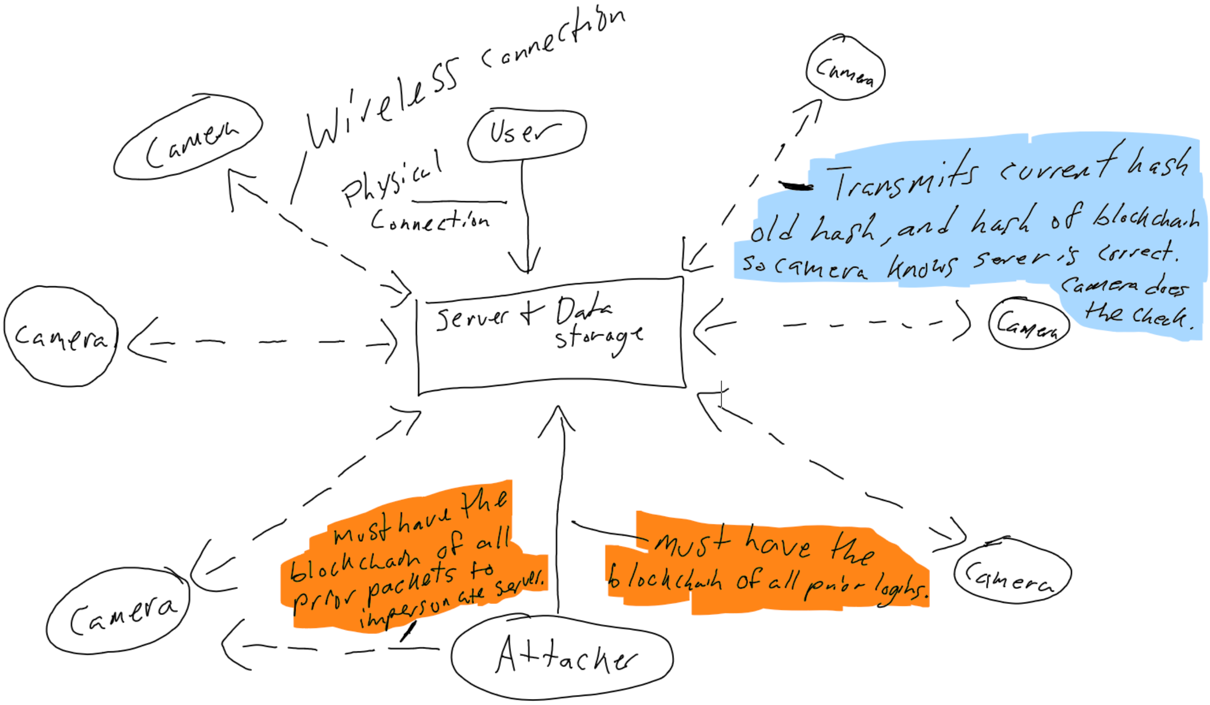
IoT systems are becoming more and more widespread, as well they are being deployed as plug and play systems, often times with little or no security on them. This can pose safety and security issues when something like a camera system can be accessed by unintended users, potentially allowing them to see what the camera sees, or even disable or change its feed. However, this can possibly be corrected by using a blockchain based system. This would still allow for minimal setup, yet have very good security.

The design I came up with is based around a home camera system. It must be able to have the intended user log in to the server to view the cameras, and keep the cameras from being tampered with. On both sides of this there will be two separate blockchains, or potentially one for each camera, with the one additional for the user to log in. They will function in a similar manner as a crypto currency blockchain for passwords, where passwords once set, must not be forgotten, and their 25 word key not lost, or the information can be locked forever. In this way, once the user sets his password, it will begin to build the blockchain with that as part of the access to it. All data transmitted can be encrypted with that as well. As blockchains tend to be more secure with more size, all login transactions and packets sent to the user can be added to the blockchain to help keep it a secure connection. If the user wanted to access the server from a new machine, they must have the blockchain with them and their password. The may make it harder to access the system, but that is also part of what makes it secure.

The cameras themselves on the other hand, will need to be paired with the server, and will begin to build their own blockchain with the server. Again using the same technique of using the packets transmitted as part of the blockchain, can both serve to log the feed, and to keep the connection secure. Within these packets, they can contain the current and old hash. To make sure the connection is authentic however, either a separate hash may need to be transmitted, that is the hash of the blockchain in its current state, or that can be the new hash. This will be the check the camera can use as it will be able to know if the server is authentic by the hashes it receives back from the server. If they deviate, this will point to an attacker trying to access it. The camera will disconnect from that source and attempt to regain the original server. The camera will only be connected to one server at a time, whereas that server can send the feed elsewhere if needed on a separate blockchain to maintain this type of security. The noted issue is one where the hardware needed for this may make the system more expensive, as the camera will need its own storage and compute power sufficient to keep up.

Once the transactions have started, if someone wanted to access this system, they must have the blockchain. If they do not have the whole blockchain, it will cause their hashes they sent to deviate from the expected hashes, and so the system can close the connection. The more of the blocks there are, the easier it will be to know what hashes to expect and the more secure it becomes. This may require a setup process where the user remains logged in for a certain amount of time or logs in a certain amount of times to build the chain enough where it can be effective in the future. The cameras should be able to obtain a secure blockchain as they should always be connected and communicating. This allows for them to be able to be deployed without worry shortly after the pairing process, if not immediately. This process may be one where the camera just needs to be physically accessed at the same time as the server till they are connected. In about all cases, this is not such a useful solution if physical access is obtained of the device, but should prevent tampering otherwise. By using blockchain in this manner, almost all connections could be able to be secured. IoT may need to have the devices be a bit more expensive till the hardware to be able to achieve this is less expensive, but this will come down to the debate of how much to pay for security.

Here is the original diagram I developed for this methodology:


Wireless connection
Physical Connection
User
Camera
Camera
Server + Data storage
Camera
Transmits current hash old hash, and hash of blockchain so camera knows sever is correct.
Camera does the check.
Camera
Camera
must have the blockchain of all prior packets to impersonate server.
must have the blockchain of all prior logins.
Camera
Attacher