PREVENTING FORGERY & COUNTERFEITING OF CERTIFICATES AND LICENSES - SUGGESTED SOLUTION

Additional search Tags: 70RSAT19R00000002, DHS, HSHQDC-16-R-B0005

The U.S. government is finally dipping its toes into blockchain projects and is getting on board with new ways to offer secure travel documents and the like. Having written previously on the topic — and having done similar projects around supply chain — we wanted to offer some suggestions.

The Request for proposal (RFP) document offers $800,000 for anyone to develop a new way to improve on existing secure documents, for things like travel documents, certificates, licenses, etc. The ID card needs to be harder to destroy or forge, yet easy to invalidate when necessary. They would also like to use the ID to link to secondary data beyond the basics of ID, yet also keep it secure from being abused.

We've written a technical paper on what needs to be done. If you'd like to see the technical details of the proposed solution, please see immediately below this article..

We feel they should also add these requirements:

1) Room for anonymity
2) Document is invalid until received by the destination party
3) Document can be invalidated at any time without possession of the document itself
4) Unauthenticated access to personal data should yield no data
5) Authenticated access should yield a trace, to avoid misuse
6) Historical data should not be deleted, only augmented

We recently did a project with securing supply chain with Near-Field Communication chip (NFC) and this technology looks like a really great fit here. NFC is a feature available in nearly every smartphone and tablet these days. It is already being used for payment processing at major players such as Apple Pay, Visa, MasterCard, and others. While NFC chip alone provides authenticity and prevents counterfeiting, combining NFC with blockchain technologies will assure decentralized data safekeeping.

As recent as this past September 2018, NXP Semiconductors N.V. has unveiled a more secure and less expensive (NFC 424 DNA) chip that generates a unique/unbreakable code at every scan by phone, tablet or other inexpensive readers connected to a desktop PC.

Combining two of the newest technologies - NFC DNA CHIP for authentication and Blockchain for data safekeeping — will solve every single scenario described in the RFP document.

One can place a tiny tag inside a document that will preclude it from being copied. NFC uses commonly respected encryption algorithms built into the chip itself. U.S. passports use a similar technology, but an earlier generation which can be hacked.

Here, we would use newly developed chips that would allow one to secure documents with the utmost certainty in their authenticity. The newly developed chips are also tamper-proof.

Chips can be read by any recent smartphone. A mobile app could be used to authenticate the chip within the document and download any amount of additional data related to the document.

The solution uses standard authentication techniques paired with any blockchain mechanism to ensure data is never erased and always appended.

What's also interesting is that documents themselves will no longer need to display any personal identification. This allows for anonymous personal IDs. People can be in charge of information inside and provide a certain default access to certain groups of users (say, full medical history to any medical doctor, yet no personal information) while denying others (nosy people always up in your business) any information at all.

No longer needed are the dreaded picture IDs, like the one appearing on your driver's license, yet it's now possible to have 200 high resolution digital pictures available for say creating a missing person's report. Implementing solution in this way will not only solve the counterfeiting and forgery problems, but it will also open up the door for endless functionality that could include secure communications/messaging, tracking, payments in any currency, eSignatures, and much more, on a scale unseen before in the United States.

It even allows a secure offline mode for agencies like the TSA, so an internet outage shouldn't stop airport from functioning. Yet it precludes TSA (or any other agency) from opening up the data on everyone without their presence, consent, and abusing their privilege.

If you'd like to see the technical details of the proposed solution, please see immediately below..

Blockchain technology is here to help, as our aging reliance on Social Security Numbers (SSN) is crumbling to ashes. Today we find ourselves facing a crisis scenario with SSN numbers not unlike Y2K. The king is dead, long live the king!

PREVENTING FORGERY & COUNTERFEITING OF CERTIFICATES AND LICENSES

Additional search Tags: 70RSAT19R00000002, DHS, HSHQDC-16-R-B0005

Problem declaration:

US government is finally getting into blockchain projects and is getting on board with new ways to secure Certificates, Licenses, Attestations and similar documents. Having written previously on the topic and having done similar projects around supply chain, we wanted to offer some suggestions.

US Government Offering Up to $800K for Anti-Forgery Blockchain Solutions:

https://www.coindesk.com/us-government-offering-up-to-800k-for-anti-forgery-blockchain-solutions

The actual document that defines the request is here:
https://www.fbo.gov/index?s=opportunity&mode=form&id=0799e4c4e8c2a2eac1ba9d5700d6ff17&tab=core&_cview=0

Glossary of terms:

This article will be talking about a number of different types of documents that can be authenticated. These could be passports, travel documents, certificates, licenses, attestations and similar documents. To avoid confusion we will call those Authenticity Documents or AD.

Problem statement and goals:

Before addressing a problem, one must define it in the proper terms and explicitly state assumptions for design.

From the RFP document:

7) Can’t be lost, destroyed or forged [seems crazy]
8) Increase interoperability of (paper) documents
9) Prove the authenticity and provenance (origin) of identity document at speed
10) Ensure that the digital document has counter-fraud protections to
    a) Increase the ease of authentication
    b) Identify indicators of tampering or fraud
    c) Increase the costs to actors attempting to spoof/fake the credential
    d) Limit/decrease the useful lifetime of documents that are counterfeited

11) Ability to track risk per individual and apply industry-specific risk-based screening protocols (e.g., trusted traveler program participant, standard traveler, etc.)
12) Track affiliation with government organizations and vendors, and roles within them, down to the project
13) AD can be issued by multiple offices
14) AD can be invalidated by multiple offices
15) Multiple forms of documentation
16) Integrate with current issuance and validation process
17) Having people authenticate access to secondary data using these ADs [Take a look at 1.2.4. This is optional.]
18) Associate external data with ADs [1.2.5, 1.2.6]
19) Must incorporate, whenever appropriate:
    a) Decentralized Identifiers (Standards Development Organization - World Wide Web Consortium / W3C)
    b) Verifiable Credentials (Standards Development Organization - W3C)
    c) JavaScript Object Notation for Linked Data / JSON-LD (Standards Development Organization - W3C)
20) Holder can grant or refuse access to information for Verifier
21) Can’t tie solution to a specific blockchain technology
22) Support for Offline mode authentication

In addition, we feel these should be part of the requirements:

23) Room for anonymity
24) Document is invalid till received by the destination party
25) Document can be invalidated at any time without possession of the document itself
26) Unauthenticated access to personal data should yield no data
27) Even authenticated access should yield a trace, to avoid misuse
28) Historical data should not be deleted, only augmented

Target Programs/Audience:

- U.S. Customs and Border Protection (CBP)
- U.S. Citizenship and Immigration Services (USCIS)
- Transportation Security Administration (TSA)

Approach - Hardware:

A “live document” is a document that is always up to date. A wiki definition can be found here.

Majority of currently used ADs are made off dead trees and are anything but “live”.

A proper AD should be a form of a “live document”, to be useful.

Majority of popular methods to secure an AD are:

- Printing process techniques, like reflective paint, additional material inserts (harder to physically copy).
- Codes that can be authenticated via a lookup

Both get outdated with time and can be copied en masse. And available documents can’t be labeled as invalid unless they are in the possession of the validator.

So for future of document authentication, we need to rely on a method which provides continuous authentication and proper encryption to do so.

Enter NFC.

Overview

NFC tags consist of a tiny chip, size of a pinhead and a substrate (e.g: paper or plastic) they are mounted into. It requires no external power and has a lifespan of about 50 years - longer than most documents today are expected to survive. With external wires that can run throughout the substrate and track if any of them get broken, to detect unwanted manipulation. It can be inserted into the middle of any printed document with no impact on shape, size, and use.

So a standard AD can be manufactured to match existing ones (for backward compatibility), but now it will contain a small chip and be able to do a whole slew of new tricks.

The NTAG 424 DNA is architected to provide AES-128 cryptographic operation. Using their “SUN authentication mechanism” - **unique** code is created upon **each read-out** by an NFC reader using the PGP keys that an Issuer has supplied upon the document creation. The keys cannot be read or modified once the document has been issued. This enables the most advanced product and content protection, plus secured and unique user experiences, served in real-time.

“TagTamper” feature that reads the state of the wires throughout the substrate go further and facilitate a status-aware, relevant consumer messages once the tag’s seal has been broken. Making tampering that much harder.

The document can be enveloped or fully made out of lightweight plastic to protect the document from being damaged by liquids or drowning in water by itself. For documents with rough lifestyle, they can be made out of harder materials, or having harder-to-destroy materials volven into them. They can also be made out of hard to cut or from fire-retardant materials.

Features

NTAG 424 DNA contains enough memory to encode a URL and some minimal information necessary for offline authentication. And some protected data that can’t be read externally, which is where the encryption keys are stored. Chip implements Standard AES-128 cryptography for authentication/secure messaging. AES-128 cryptography is used by governments worldwide for authentication and secure messaging. In addition to the standard implementation, there is an optional AES-based protocol that uses a Leakage Resilient Primitive (LRP), which increases attack resistance even further.

Chips can be read by any standard NFC reader that supports the right frequencies, which includes almost every smartphone manufactured in the last 4+ years.

This means using industry standard hardware across the board, for ultimate platform flexibility.

“Each time the tag is tapped, it generates a Secure Unique NFC (SUN) authentication message using an AES-128 cryptogram. An NFC-enabled device reads the tap-unique URL which contains the encrypted code, sends it to the server for authentication, and returns the verification result. The SUN mechanism is compatible with all standard NFC phones, including Android and iOS-based phones, and offers a more secure way to maintain data confidentiality and integrity.”

Chips can be “locked” after information has been written, to avoid any further modification to the data.

References:

https://www.nxp.com/docs/en/brochure/NTAG424_INDUSTRIAL_BROCHURE.pdf

https://www.nxp.com/products/identification-and-security/smart-label-and-tag-ics/ntag/ntag-for-tags-labels/ntag-424-dna-424-dna-tagtamper-advanced-security-and-privacy-for-trusted-iot-applications:NTAG424DNA

Approach - Client Software:

To facilitate this, we would need a secure device running android or apple IOS, coded to a secure app store. This is a standard now for all military devices, so this is nothing new. Simply locks the device to only run authenticated software, authorized by the governing authority.

Now that we are guaranteed the right software to install, we have a secured application loaded on the device with NFC reader (tablet/phone/whatever).

All communications with server happens via full bidirectional HTTPS protocol to facilitate encryption and guarantee that the server is properly secured and so is the client.

Scenario 1: Document Use/Authentication:

Holder (Bob) approaches Verifier (TSA) and hands over the travel document.

Verifier touches the document to the authentication device's NFC reader, while running authentication app.

The app reads the secure URL which contains encrypted substring and passes it to the server for authentication. Device itself does not need to contain any information needed to decrypt these messages (unless FULL offline authentication is required) which significantly increases the security of the process.

Server authenticates the chip and returns back the information about the traveler. Including any amount of additional information required by Verifier, and is limited to their role within TSA. Additional features possible here are a temporary 1-time-use passwords that Holder can generate for the Verifier to facilitate their access to additional information, or to ANY information, depending on configuration (more information in a lower section).

Verifier's demand for information is registered against their own login by the server infrastructure, so it can be monitored for abuse.

Additional information can contain things never visible to the Holder themselves and can be used for measuring their risk, and allowing Verifier to send additional information back for tracking by future Verifiers.

Scenario 2: Document Creation:

Issuer contacts any manufacturer capable of handling an order of 10,000 forms with embedded NFC 424 DNA chips. Manufacturer does not possess the security keys that Issuer uses, so these forms being lost or stolen or duplicated at this stage have no security threat. Documents can also look and feel different between states/business/locations/tribes, but must only contain the right NFC chips.

Holder (Bob) approaches Issuer (TSA) with acceptable proof of identity and demands a new "travel document". Proof of identity can be 5 other people with existing "travel documents" along with standard issue passports etc.

Issuer verifies the Holder, assigns them an account ID (based on the blockchain's underlying AccountID), grabs the document with embedded NFC chip from the pile and runs it by the NFC encoding device. Device encodes current encryption keys to the NFC (keys can be generated locally or remotely and work like tokens and can be valid for next 5 minutes only), NFC is programmed with any additional information and locked from further writing.

The keys are sent to the backend along with NFC device ID, which creates a record in the blockchain of choice that the Issuer is using. And upon this transaction completing uninterrupted - the "travel document" is created.

Any of these steps do not complete and the document is void.

Document is then sent to the Holder with a temporary hold on the account. Once received, Holder can authenticate with Issuer that they received the document and the document becomes active. This avoids document theft or misuse prior to reaching the target Holder.

Scenario 3: Document invalidation:

Upon learning the documents have been lost or damaged - Holder notifies Issuer of the fact.

Issuer adds a record to their existing blockchain's account records for this Holder that old ID is invalid. Any further inquiries against the old IDs will return a message to Verifier to confiscate the documents and destroy them. (This avoids misunderstandings as the true record is the blockchain, so one can see who demanded the documents are invalid. So physical destruction is prompt, warranted and authenticated.)

Immediately or at a later date a new ID is issued, and a record is added to the same account sequence that identifies the new ID and keys issued to the Holder, so the complete chain of events can be maintained. As you can see: the procedure for Document Creation is repeated with minor changes.

Scenario 4: User-Secured Information Requests:

If the architecture is to allow Holder to restrict the distribution of their information, it can be easily accomplished by allowing Holders to generate their own private-public key pairs. Public key gets submitted to the Issuer. Holder then uses their private keys to generate temporary short-lived one-time requests to allow AGENT#123 to see certain data associated with their profile. Data can be stored locally on Holder's device or with Issuer.

So anonymous travel documents are even possible, if desired.

If PGP is an overkill, Holder can simply use an app on their own phone paired to the document's NFC key (in the same document) to generate a password that expires within say 1 minute which grants Verifier's access to some/all of the information.

Holder should be able to request a log of access records against their own document using their phone app to see who has accessed their personal data, when/where.

Issuer can still add an interpreting layer that reads certain flags on records and not show those records to civilians (for say military personnel).

Scenario 5: Offline mode:

Offline mode would not be horribly different from what is used now, where agent without internet can still verify the printed information against Holder's other documents (like credit card and driver's license).

But now additional methods are available:

1) NFC chips can store basic personal information, like person's name. So, in absence of internet, agent can ensure information in NFC storage matches information printed on the ID.
2) If we allow for Holders to generate their own keys, as Scenario 4 describes - the public keys can be cached on Verifier's devices (as we are only storing public keys, which are tiny and can be further set-reduced to reduce space requirements) so that verification is possible on the spot without internet, by having Holder encrypt a password the agent gives them and being able to verify it immediately by agent without using internet. One can use QR codes to expedite this process.

Approach - Serverside: What happens on the server:

The Issuer maintains a blockchain of choice. If blockchain contains a virtual machine, some of these server tasks can be shifted to the blockchain for increased security. Need a blockchain that can at least store some value pairs securely. Doing this with a minimal blockchain like Bitcoin would be much harder than with something like Ethereum. Best design would have both, a virtual machine and Zero Knowledge data storage.

Blockchain will contain accounts for each Holder and each Agent (person allowed to use the system), under which transactions will be listed, and either references to external data or the data itself will be recorded on the blockchain.

Issuer can also run an Oracle for decrypting data, fetching external information about the Holder not available on the blockchain itself.

Issuer would also act as a data aggregator. The use of blockchain addresses to fetch data would also help with keeping the externally stored data somewhat anonymous.

All the information needs to be encrypted, to allow for security of the keys and other personal data. If someone gets a copy of the blockchain nodes - theys shouldn't have any actionable data at their disposal. Oracles and another API can be used to facilitate this cleanly.

Example document creation:

Issuer's Agent (Account AGENT1) logs in to the system at device DEVICE1.

Blockchain stores both facts against AGENT1's account.

AGENT1's account role and permissions are read. AGENT1 is allowed to create Travel Document and nothing else. This can be verified against every action AGENT1 performs against the blockchain and any actions outside of that role will be disallowed and immediately logged and reported.

AGENT1 gets a request from Holder that they need a new Travel Document.

HOLDER1 account is created and Holder's information is assigned to that account.

Travel Document ID is sent to the Issuer's microservices (via API) with a request to associate it against HOLDER1's account. Blockchain is asked to run an atomic transaction where it will associate the two if no other association exists. Part of the request are the security key ID assigned to that Travel Document ID. Keys themselves can be stored here or just their ID, depending how key storage is managed. If keys are stored, server uses server-side encryption to secure the keys themselves so they don't get stored in plain text. A higher level security can be used, like AES-256 or 512 as this storage is high risk. AGENT1's ID is stored with the record, along with date and time.

Follow similar logic for temporary hold: account has a flag of "on hold" and requires for another agent or an online service to remove the hold once the docuemnt is received. At which point a new record gets written which says "AGENT2 certifies this document to be received, date+time".

Example document invalidation:

AGENT3 (or some API) receives information that Holder's document should be invalidated. Upon verification, we add a record to the account HOLDER1 that says "INVALID DOCUMENT", marks involvement of AGENT3 and date+time.

Any requests against same Travel Document ID or against HOLDER1 will return either all records for that document including "DOCUMENT INVALID" or a summary record which will simply tell AGENT4 that this document is invalid.

Example document validation:

AGENT4 uses their application to send a request for data using the key retrieved from Travel Document that Holder gave them.

Records gets added to the blockchain that AGENT4 is examining data for HOLDER1 account retrieved using DOCUMENT1 at geo location matching AGENT4’s known and authorized location of ZipCode12345.

System retrieves KeyID of the encryption keys used for DOCUMENT1 upon creation from blockchain.

Verification URL passed in from the verification device contains a message that’s encoded with AES-128 encryption using KeyID we just retrieved.

Without fetching the keys themselves, an expert system is requested to retrieve data based on key’s index number and the string provided. Message is verified and expert system returns information that either supports that DOCUMENT1 is legit or not.

AGENT4's profile is retrieved to determine their role. Based on that and which API call was made by the agent's application - additional information is retrieved and appended to the reply, to ensure only the information AGENT4 is allowed to see is shared.

Another record is stored on blockchain with the same answer and a copy of the query response is sent back to AGENT4’s app.

Oil pipeline problem [1.2.5, 1.2.6]

Say we have pipeline running through backyards of DRILL1, DRILL2

And we only want to allow oil from DRILL1?

And assume zero trust. And that DRILL2 can make a deal with DRILL1 to also dump oil into the same pipe and have DRILL1 say it’s their oil. Then DRILL1 will only pay them 90 cents on a dollar and confirm they are producing the oil that DRILL2 is dumping into the pipe.

This is somewhat unrelated to the earlier but can be solved via a blockchain as well. We will be applying supply chain solutions here.

Solution 1: Spectrometer

One can use spectrometer (or similar) to document the chemical composition of oil (or raw minerals) produced in a particular area. Other spectrometers can be installed in the pipe so as the composition changes - readings and their locations will be sent to the central facility. Where they can be verified against the blockchain that records who is producing oil and when.

Even if the oils mix - it should still be possible to verify possible and impossible combinations based on number of indicators present in the oil. Then investigations can be launched when there is a mismatch.

As chemistry of oils from DRILL1 changes over time, it can be tracked and updated.

We do not suggest a use of blockchain for all these pieces, but only for keeping records of authorized agents as they interact with drills, owners and pipe, in a manner similar with tracking records for the other portion of this document having to do with record keeping around documents.

Solution 2: NFC Tagging of materials

We can also use NFC tags inside neutral buoyancy substrate to travel along with the oil. And these NFC tags can be encoded by the original oil supplier. So that as the pass through tag readers inside the pipe (or we read them when we fish them out at the end of the trip) - we can verify their authenticity, and only pay out on the ones which are authentic.

Similar can be done with shipments of natural resources like raw materials. NFC tags can be attached to the containers and to verify containers have not been tampered with (using TagTemper technology). Then can be scanned at ports and verified in authenticity. Records created on the blockchain by scanning at each port will increase the validity score of the information as well.

Solution 3: NFC Tagging of people

One can use NFC tags for badges and require that badges be used to sign into the applications where the deliveries are being recorded.

This decreases fraud and guarantees that a stolen password can’t be used to submit false information. Also, requirements can be dictated on readers to provide geo information. So information can be attributed not only to the person, but also to location.

As earlier - all interactions can go onto a blockchain to reduce fraudulent record modification.

Possible attacks

Previous generations of NFC tags relied on things an NFC manufacturer should and shouldn’t do. As well as “security through obscurity”. The new Tag 424 is finally a generation worth using.

As keys are used to generate an AES string and NFC memory for storing keys is inaccessible through the NFC’s own API, the only way to read that data is by possibly using a very serious microscope or other technology only available at extreme costs, if at all.

The rest of the system uses modern algorithms based on PGP and is only vulnerable to Quantum computing or applying huge computing resources to defeat a single key. Making the cost extremely high for an attack to work on a single NFC chip. And making it extremely unlikely to be used at scale.

As manufacturing of NFC chips catches up - one can increase complexity by going to larger-key algorithms like AES-256 and higher. WIth each bit added to the key - complexity doubles, making brute force attacks useless.