February 28, 2014 Meeting Minutes

- Minutes approved in 12 March 2014 meeting

Role Call

Roll call taken by Bob Griffin. Quorum was achieved.

Proposed Agenda

08:30 to 09:00 - arrival at Bloomberg facility, start concall bridge and webex, et

09:00 to 9:15

Roll call Agenda Minutes from previous meetings (22-Jan, 5-Feb, 19-Feb)

09:15 – 10:00 – v2.40

review/resolution of any outstanding issues for v2.40 docs possible vote to initiate 2nd public review status of Statements of Use,

10:00 to 10:30 – review/resolution of issues regarding substantial additions to v3.0

Stef: Function tables BobR: Parameter passing

10:30 to 10:45 – break

10:45 to 12:00 – review/resolution of issues regarding substantial additions to v3.0

Wan-teh proposals How to do more rapid update of mechanisms 2.5 or 3.0 decision (consider how much new functionality, impact on implementations, visibility in industry, etc)

12:00 to 13:00 – lunch

13:00 to 14:00– review of PKCS 11 interop at RSA Conference, possible discussion of cross-TC testing going forward (Tony)

14:00 to 15:00 – discussion of new functionality, apis etc to leverage v3.0 flexibility (Sven)

15:00 – wrap-up and close

Agenda Additions

- Secretary discussion

Motion to accept agenda

- Tim Hudson Moves
- Bob R. seconds the motion
- No objections - No Discussion
- Motion Passes

1. Approve Previous Meeting Minutes

- Minutes up for approval: 22-Jan-2014, 5-Feb-2014, 19-Feb-2014
- Tim Moves
- Sander seconds
- No objections or discussions

2. Secretary

- Venafi has not renewed their OASIS membership, so David Smith will be stepping down. The chairs would like to thank him for her service and are looking for volunteers to take over the role. Please reach out to Valerie or Bob G if you're interested. Valerie and Bob will take minutes today.

3. Update on 2.40 documents

- Chris Z.: updated all the easy things like search and replace items for the base document. It would take a long time to integrate all of Patrick's minor nits, all big things have been addressed. Spent a lot of time trying to update the current mechanism document, documents that we reference (that are old documents that places like ANSI no longer even posts). Patrick said we need to have pointers to all normative documents. Chris has spent a long time chasing these done, some may be so old that they are unfindable. He was very thorough with his review of a large document, so Chris feels compelled to do as complete of a job as possible. At this point, it would be major changes to remove old mechanisms, and Bob does not think that's a good idea at this point in time. Bob suggests that we note we will track this down and work on cleaning it up in the next revision. Bob R. asks if we can use secondary references, Chris does not believe that would meet Patrick's needs. For example, we have copies of some of these documents, but there is no longer an "official" source.
    - Bob G. would like to take time to review these documents and then vote on our next meeting to take these to public review. Bob G. will reach out to each commenter and explain our approach and thank them for their comments.
- Sue: have integrated Wan-Teh's and Patrick's comments, posted this morning. There are no comments outstanding.
- Tim: Profiles doc: Bob G. posted the version that incorporates the feedback that we were planning to, consistently with changes being made/accepted by the other editors.
- Bob: Has also updated the usage guide in a similar manner
- Tim proposed a motion that pending any final changes to the document he'd like chairs to reach out to OASIS to start the final public review.
    - Bob R. seconds
    - any discussion? Bob G thinks that's a reasonable thing to do. Didn't get to look at Chris's latest version, but did review the previous and thought we were in good shape.

    - No objections or abstenions. The chairs will submit a request to OASIS and give an update at our next meeting.
- Bob G: we'll need at least 3 statements of use. He does not believe that any one person supports
    - Tim said we have to list the conformance clause, and you have to explicitly list that you're interoperable with another Vendor's implementation.
    - Bob G: is in the process of getting a statement of use from RSA/EMC
    - Bob R: any examples? Bob G: Tim's and RSA/EMC's are good examples. 3 is the minimum, more would be nice. We'll need these in the next month or so.

4. Stef: 3.0 proposals: Function Tables

- Immediate goal is to be able to add new mechanisms more quickly and not have vendor defined mechanisms conflict with one another
- This is designed to be something that will stick with us from here on out, please speak up if you have any issues
- there is no such thing as an uninitialized function table, once you get there, it's initialized
- There will be a NEW entry point, which accepts a function table and an output context. This will not release resources until the final caller has released.
    - Let's say you want to ask for a vendor extension? You'd ask the current interface for another one.
- Function table is wide open. Will likely bring over a lot of things from PKCS11 2.4, but he has not had time to dig into that. There is room for other contributions there
- Bob G. Suggest we walk through the new interfaces, Stef agrees
- Must have a C_Release function where the caller can say that it's done, and a C_GetInterface to get another interface from an already initialized interface
    - For known interfaces, those strings would be in there.
- Valerie is confused about the interface terminology
    - Stef notes that there are the standard interfaces (C_Login, etc), but the vendor can supply additional interfaces (Valerie referred to this as a vendor extension interface). Stef said this could be things like one interface is 2.X and another interfaces is 3.X and another 4.X, etc.
    - you pass in which one you'd like (standard, vendor defined, or additional standard PKCS11 interface)
- Michael St. Johns: this sounds like a reasonable approach for an API being incorporated directly, but how do you play in an environment where it's an extension, like Java?
    - Stef: if there was an API that could be exposed to better interoperate with another API. There's nothing here for abstractions, like python bindings. It needs to already be known the calling application. Michael: that answers the question, but argues that we should be looking at this in two different ways.
- Tim: Over years, vendors have all added their own extensions. Stef has come up with a way to incorporate what the other Vendors have already done. People extend Java all the time.
    - Michael: I didn't say that. I use non-public functions when I use Java, because I need functionality that is not generally available.
    - Bob G. : Does the concept of the initialize sound okay to you? Michael St. Johns: not online, can't look at the moment, will look when he gets a chance.
    - This is very low level functionality, there's other crazy stuff you can do. We will need additional proposals to fill this in and do the things that Michael wants to do.

    - Tim: I think language bindings is a separate and worthwhile topic, but it's not a part of this discussion. If want a nice smooth Java binding, we need to look at that separately. Mike: I'm not asking for a Java language binding, but I'm saying the languages out there treats PKCS11 as a plugin. It's an exposed functionality and setting those expectations.
- Back to Stef
- In this new interface, there's no C_Finalize. When you're done, you call C_Release. Releasing the function table that you used when you called the initialize
    - Look at C_GetInterface, a method to get additional interfaces.
- How does this interact with 2.x?
    - The entire workflow is different. Many of the concepts remain. This should help with migration. It allows for multiple interfaces, so that this can be leveraged.
    - Initialization should be thread safe.
- Bob G: Any issues you saw for not doing an interface definition for 2.x? Stef: Trying to put this in caused too many ugly exceptions. It makes more sense to make 2.x and 3.x distinct. That puts a lot of effort onto the module vendors, by supporting 2 APIs. This can be handled with SHIMS.
    - Bob R: The only reason you'd have to have 2.x as well was if 3.x required it. This would be for an old consumer using the old interface. It's not as pretty, but that's not how it will be used.
- Valerie: it would not be mandatory, then, for a new vendor to put in 2.x?
    - Stef and Bob R. and Tim: Not mandatory , but vendors may choose to do so to offer the most options.
- Tim: Have you prototyped this?
    - Stef: Not yet, want to prototype the shim with a mock interface to make sure this actually works.
- Stef, in reply to Bob G: I want to work on the shim first to show how it works. this will produce 2 parts of the spec.
- Sven: I find it's dangerous to discuss items like this until we have defined what exactly should be in 3.0, which relates to this afternoon's conversation.
    - Bob G: This relates to anything that we wan to do for 3.0. We will have this problem
    - Sven: there may be proprietary commands, there might be different commands or just different syntax?
        - Tim: a mix of things. All of the above. At our first meeting, I showed vendor extensions. Some vendors have 100s of extra functions.
        - Sven: people are trying to lock people into their implementation. Perhaps we should define those extensions as standard?
- Sven: if we want to say that we can only add things in 3.x, we will be extremely limited. We could gut everything, we could standardize, different distinct sets of optional functionality. Here is how to enroll via the web, here's how you enroll with these types of devices. In addition, we then will have an example to vendors on how to extend. Previously, the way vendors extended (like adding something to the end of the function table) prevented the [technical committee] from extending. On it's own, this proposal is nothing - this enables the addition of new things.
- Bob R.: this basically says we are going to add new functions to 3.x - how do we do that safely? we can add 2 new functions or 50, but we'll still need to do this. there is a hidden (or not so hidden) feature: he's working on a way for different modules to use one library in a shared address space.
- Bob G: when will you have a shim?
    - Stef: 4-6 weeks

- Tim thinks this will take us in the right direction.

5. Bob R: 3.0: Parameter passing

- How do you store meta-data about PKCS#11 libraries: which ones to load and where they are. A good place to store config information (like which mode you were in)
- In 2001, there was a proposal for some of this, like using strings and namevalue pairs. Initially defined:
  - library=path (location of module)
  - name=name (what to call this instance of the library)
  - parameter=free form string (an unparsed string to pass to the library)
    - could be vendor specific, like NSS=
- the application reads this in at the start and loads things in turn
- Current valus in the NSS= line:
  - Flags like internal, fips, critical, moduleDB (fips==FIPS mode)
  - TrustOrder (who has precedence for validating cert)
  - CipherOrder (who should have precedence for doing which cipher, softtoken for them has lowest precedence by default)
  - Slot params
- these are stored in a file. there is a standard location, but can be overridden by an application. A vendor can add their implementation to that file (some are , others are not).
- valerie: solaris has something similar, but called pkcs11.conf. Similar things, different format, but can be extended. We also have policy (like admin can disable a mechanism per library or system wide, like MD4). Bob R: We want to add policy to PKCS#11 standard, Solaris could drive this as a reference application.
- Tim: Java also has a configuration file for PKCS11 attributes.
- Stef: there's a reason we'd return a different function pointer when C_Initialize is called twice: you could send different algorithms, like a different configuration file. if they don't support it, that's fine. If they support multiple callers, they could return the same pointer every time if they wanted to. You can also set it up so they get a different pointer, s there are no conflicts. Could have different databases for softtoken setup.
- Bob G; What are the next steps? non-normative usage guide? or would this be normative as a new kind of information in the spec? Bob R: new information in the spec.
- Bob G: would this be a problem with people that already have one?
  - Valerie: ours is called pkcs11.conf, not pkcs11.txt (Bob R agrees .conf is a better name) , but that it's "consolidation private" (ie not a public API). We would have to work with a few vendors, but not the general public on changes.
  - Mike st. Johns: you're not planning on changing C_Initialize right now, are you? Bob R: we've changed the arguments in NSS, but we're not covering that now. Mike: other people are already using pReserved
  - Stef: we should discuss one file vs several.
    - Bob R: this is an old proposal, old link is still there. What do you do when you have a system with 2 archs on one system: 32-bit and 64-bit binaries on one system. Can't load a 32-bit binary in a 64 bit program, bad times. This can get more complicated cross platform (ie Solaris can't load an AIX module). In NSS, if something doesn't load, it will be skipped (unless it was marked critical). Valerie notes that Solaris is similar.
- Stef: may want a directory for files

- Wan-Teh: complaints from co-workers: they do not like the syntax, but they are a generation that started learning with Python first. String format needs to be defined. XML is another alternative, but it is worse. Other complaint: no C function to build these strings (Bob R: this is fixed, we've exported our internal string building functions)
- Sander: not sure what happens if we initialize with non null pReserved? Tim: it goes bang. Sander: we have our own configuration that operates outside the calling application, we do not accept configuration information from application. Tim: We need mechanism for application to pass it in, or have implicit pass in (from standard config).
- Bob G: let's define this over the next month or so.
  - Valerie: I (or someone from my team) can help with this, Sander would Thales want to help, too? Sander: sure.
- Bob R: more info Https://develoer.mozilla.org/en-US/docs/PKCS11_Module_Specs
- Action Item: Valerie, Bob R and Sander: work on configuration information.

6. Wan-teh: 3.0 proposals

- We need to encrypt several messages using one key. Not just our need, SSH and IPsec does something similar. We used to only be able to do this with block ciphers. would like to add this for things like RC4 (stream ciphers).
- We were doing 3 PKCS11 calls for one TLS record. Is there a better way? Is it possible to reduce this and lower the overhead of a PKCS11 call, esp. for a software only implementation.
- could we do something like C_OpenEncryptAssociation: encryption mech, IV generator mech and key, then encrypt as a single operation?
- there are alternatives to consider. Things cannot always be verified. For example, if the module needs to return a failed status, it must not reveal any decrypted plain text - difficult to do with stream ciphers. One way to handle this would be to only allow single part mode for decryption.
- All AEAD algs Wan-Teh has found, they require all of the AD be available before they provide the first byte of plain text.
- Should we have a separate tag argument? only S/MIME works this way.
- An IV generator mechanism specified at the same time as the encryption mechanism? same lifetime as the encryption op, but not applicable to decrypt
- Ideally existin echanisms also work in the message-based mode.
- Bob R: Is the signature of C_OpenEncryptAssociation the same as C_EncryptInit?
  - Wan-Teh: depends if you want to add the IV generation mech
  - Bob R: I think we'd want to, otherwise there's no reason to have C_OpenEncryptAssociation
  - Wan-Teh: depends on if you want older mechs to be able to use this
  - Bob R: the generic (?) parameter has served us well. If we had this already, we wouldn't need this proposal for AEAD. other things may come. Do we want to minimize our functions? if we do, let's think about if there is a semantic difference between this and C_EncryptInit let's not introduce a new one.
  - Tim: I don't see a semantic difference, either
  - Wan-Teh: I think there's a better way to decide how the mechanism should behave based on the path it takes
  - Bob R: then that's a new mechanism, not a new function.
  - Wan-Teh: there would be parameters ignored by the new function

    - Bob G: Does this address Mike's comments?
    - Wan-Teh: I believe I've addressed Mike's concerns. I may not have made the same choices Mike wanted, but have tried to address them all.
- Bob G: any comments on the phone? none. Then the only issue seems to be mechanism vs. new function?
    - Bob R: sounds like a vote might be needed
    - Tim : I'm cautious. want to stick with what we have unless I see the reasons for this approach
    - Bob G: can you do a short summary email: pros and cons? either a straw poll or we look at both proposals and vote on them?
    - Wan-Teh: I can do that
    - Bob R: want to see a straw poll
    - Sven: is there a practical example?
    - Tim: this is a style thing, I think.
    - Wan-Teh: practical example: large number of packets need to be encrypted with the same key
    - Bob R: and not lose context.
    - Chris Z: we also want good performance, will need IO vector, too.
    - Bob G: will set up a straw poll on this, to help give direction.

7. Wan-Teh: Pseudrorandom for PBKDF2

- proposal of new flags.
- Bob G: We could consider whether we add this to 2.40, if it's of small enough import. Bob R?
    - Bob R: this was the case I was thinking of, it should be easier to extend w/out going through the entire OASIS process.
- Bob G: two major options, just add it in before our next public review (would need to check with Chet). could we do this in a 2 week review? or maybe 4 week?
    - Tim: We cannot pretend this is a non-technical change, OASIS would call this a substantial change.
- Or, we keep it frozen and do a quick 2.41 or 2.5? This is the only thing Bob G has seen so far that is high priority that we should consider.
- Chris Z: take the vote
- Tim: haven't we already left some mechs out of our public review 1?
- Chris Z: not sure when I added them.
- Tim moves to add this to 2.40 to the table and the current text into the current mechs, with the proviso that this will not have significant issues with our 2.40 review with OASIS.
- Bob R seconds
- No objections, Bob G will contact Chet.

8. Forking to help with forking ?

- Bob R: a semantic we agreed on for 2.40 should help the vendors with this. Applications do not always know they are calling NSS, for example (as another library calls it). Forking is problematic in this case. Is this a hardship on hardware vendors to handle this?
- Bob G: anyone else have this problem?
    - Valerie: yes, we have libraries that call our pkcs11 impl, so we tell library devops

to NEVER call C_Finalize or bad things happen.
- Bob R: small change to the documentation, but bug change to the implementations.
- Bob G: may need to gather some more requirements before we decide to fork the standard

9. PKCS 11 TC Interop

- summary of Interop (Tony). Included both KMIP and PKCS 11 TCs, showing a number of PKCS 11 implementations. Less formal on the PKCS 11 side. Feedback very positive from OASIS, booth staff and visitors.
- Tim: range of conversations from high-level to detailed, with exceptional cooperation among vendors. WOuld be good to show more interoperability in the future, as with KMIP. Included products from vendors who weren"t yet in the TC
- Valerie: ve4ry valuable for sharing information about the move of the TC to OASIS, etc.
- Graham: particularly valuable as a new member of the TC. Showed a lot of interest.
- Tim: very valuable to have co-chair in the booth. OASIS already looking for interest for next year. TC has to decide whether to do an interop next year, including getting test cases defined, etc. Challenges when trying to move ot larger range of environments. Being present at the show helped people realize how many people are using PKCS 11.
- Valerie: a lot of vendors and users came by.
- Tony: included very broad range in terms of organizational roles, companies, etc.
- Sven: would have liked to have clearer message about the value of PKCS 11, what it does, etc.
- Tim: incredible mix of people, but generally one consistent message: "PKCS is going places". Number of vendors, interactions etc was exceptional, much more extensive than first KMIP, for example.
- Tony: very valuable in increasing visibility and membership for OASIS, then helpful in terms of TC visibility, then finally in terms of the standard. Also valuable as an internal marketing exercise.
- Sven: lot more to PKCS 11 that should be communicated, particularly the non-technical things.
- Tony: most people stop by for just a short time.
- Temmer: was placement of the booth beneficial in terms of traffic?
- Tim: definitely better than last year
- Magda: was able to talk about PKCS 11 actually doing something, lots of people asking about PKCS 11.
- Valerie: very good idea to have the HSMs etc by way of simulating interest.
- Magda: comment that the interop was actually something that worked
- Tim: incredibly busy week.
- Magda: Both Apple and Microsoft stopped by
- Valerie: talked to a number of people from Apple. May be people in the company using it.
- Tim: there are definitely people in Micrososft who are interested in PKCS 11.
- BobG: should we start looking at test cases?
- Tim: first step is to look at whether the TC wants to be at RSA Conference or not.
- Tony: this year did not focus on common test cases. Would be good to look into defining them.
- Tim: can start with the small step of profiles that have been definied.
- CHris: what is Google position regarding Android etc.

- Wan-teh: most Android crypto operations are performed by Java. Prefer to use crypto native, using PKCS 11 only on Linus.
- Sven: as software vendor, portability of interface in PKCS 11 is valuable.
- BobG: what about addtional profiles, test cases, etc. Might be valuable in testing out the design for new functionaliy?
- Tim: this will probably be version independent.
- BobG: any plans for prototyping of new P-11 functionality by way of understnading possible design issue?
- Wan-teh will be prototyping AEAD

== 10. PKCS #11 futures (Sven)

- Discussion of competing crypto interface standards (Sven). Significant driver in terms of change in end-user devices (laptop to mobile) raising interoperability issues. We should consider moving P-11 from simple crypto interface to a more universal solution. For example, Microsoft new virtual smart card, etc.
- Tim: question is which devices can't be handled in P-11 at the moment?
- Sven: exactly the point. It offers the opportunity for more universal use of P-11. Example of Microsoft messages (see Sven-s slides).
- Tim: convenience of the Microsoft interface is how it's been packaged. Not an API issue, but delivery on a platform.
- Sven: saying that P-11 is an important element that needs to be enhanced.
- Sven: discussion of P-11 as not just a token driver (see slides). For example, possibility of user using same badge across all platforms, compared to current need for different smart cards for different devices.
- Dina: at Oracle, have started devising a more likely interface that gives direct access to the crypto functions with less overhead than P-11 (micro crypto interface). Perhaps PKCS 11 lite?
- Chris: to some extent a perception problem: most of the time spent in spent in implementing the application.
- Dina: significant difference in throughput comparison between P-11 and lighter interface.
- Tim: have seen both fast and slow implementations
- BobR: depends on things like your session design, rather than the processing cost of calls etc.
- Sven: Microsoft had to new generation (minidriver) in order to resolve the problems they had with a weighty interface.
- Valerie: there was definitely start up time, etc.
- Anthony: working in Java, can see a lot of issues just in terms of the interface design in P-11.
- BobR: problem is most noticeable in smaller apps/processing.
- Anthony: it's the cumulative effect. Also, issue in terms of developers getting started.
- Valerie: want to see what we can do in terms of combining things into init, etc.
- Chris: part of the issue is fewer cookbook examples for P-11 than for OpenSSL. P-11 not intended to be all things to all people.
- Sven: is there a possibility of reviewing the api to make it easier, better performance, etc?
- Tim: also need to discuss what we're trying to accomplish, certainly differs across the TC
- Chris: we did consider some of this in the earlier discussions.
- Sven: one of the topics to discuss is what we call the next release. Should it be 3.0?

- Tim: had a good consensus that the significant functionality warrants calling it 3.0.
- Sven: comparison of PKCS #11 to MS CAPI, tokenD and Blackberry API (see slides). Also brief touch on Android.
- Sven: comparison to competing card APIs: .NET, PKCS #15, G&D initiative on Android (see slides).
- discussion of issues from Charismathics (Sven) (see slides)
    - support for multiple pins.
        - Tim has this been solved by other apis?
        - Sven: definitely a customer requirement, example of limitation in PKCS 11 to support this. Virtually every card uses a different approach.
        - Chris: has been an issue for BloombergSven
    - UEFI support
        - Sven: UEFI allows import of data from pre-boot to post-boot, would allow avoiding double entry of passwords during user login
        - Tim: can see value of single credential through boot process
    - biometric interfacing
        - Sven: currently no way to shift biometric information though P-11 to token. Could help to bring data to app level for comparison.
        - Chris: also have this issue, addressed with hack to C_login
        - BobR: need to get people who have tokens like this together to get a proposal together
        - Sven: can bring in the biometric vendors, but could be difficulty with competitive issues.
        - BobR: API should be different question from proprietary data formats
        - Chris: dealing with authentication for biometrics should not be in P-11.
        - BobR: but do want a single way to ask them for a yes or no
        - Tim: FIDO is trying to do this
        - Chris: have worked with most of the vendors in this space, but best to stay out of it.
        - Sven: idea was to have P-11 be the simpler interface for the vendors
        - Tim: need interface to allow vendors to plug into.
    - Predefining additional cert parameters to differentiate certs (selecting certs or apps by parameters(
        - Sven: currently done in proprietary ways, not sufficient detail in X.509. INcreasing need to be able to distinguish certs for different purposes, as well as to specify what apps a cert can talk to.
        - BobR:would like to have Stef look at this, in relation to questions like "What is the cert trusted for?"
    - Feature list of addressed token and supported functions
        - Sven: to understand what a given version of P-11 supports, probably addressed by function list functionality
    - Clear and granular error messaging
        - Sven: problem of unclear messages and of not being able to catch some messages
        - Valerie: many current error codes are very hardware-specific.
        - BobR: have an example of this.
    - Deployment handling features (token init, app loading)
        - Sven: significant issue because of customer need to move from one token management system to another. Small addition to P-11 could help.

        - Graham: is this a goal of Global Platform?
        - Sven: is there a way of focusing on commands to simplify the process
        - BobR: coudl be just for loading the token?
        - Tim: Need to define exactly what we want to address, compared to platform-specific
        - BobR: Red Hat uses initialize_token for soft tokens. Not sure how widely the call in used.What we need is exactly what the card managers want to put in here.

Some of these have been recognized, but did not have the right people/companies to address the problem.

- Tim: could look at how to bring Global Platform and P-11 closer together.

* Security gaps (Sven)

- data transfer encryption.
    - Sven: need to look at the unblock. Currently proprietary solutions, would like to offer something cross-platform
- code vulnerability in operation
- secure PIN entry
    - Sven: example of virtual keypad. Is there a way that we can serve this feature better?
    - Chris: TIC issue let to Bloomberg not using virtual pinpad
    - Sven: key point that PKI is considered difficult; through offering optional settings to customer we could help make it easier to use

- Technical issues and architecture
    - cross platform PIN policy enforcement
        - Sven: max and min pin length should be configurable after initialization.
        - Sven: also issue of differences between numeric and alphanumeric
        - Valerie: could apply the configuration enhancements we discussed this morning
    - defining critical UIs
        - Sven: some features may require specific UI capabilities. Can P-11 help in managing what UI capabilities are presented?
    - application authentication procedure
        - Sven: competing standard is ISO 24727, to authenticate application against the user. Can we offer anything that addresses authentication of application? Perhaps certification process?
        - BobR: issue is that problem can be application, rather than library.
    - application conflicts and slot management
        - Sven: issue of locking module. Matter of clarification?
        - BobR: Stef is working on this with regard to Find_Function.
        - BobR: NSS addresses the issue of slot management through using Find_Objects. But may be able to help in terms of search.
- Sven concluding remarks
    - Importance to think about what we anounce, particularly in terms of making P-11 much more available and convenient.
    - BobG: we should consider whether any of these help of us that regard, especially in addressing the possibility of interoperability

  - Tim: looking at level of interest, etc helps us to sort out that question
  - BobR: we should at least look at larger issues, like multiple PIN issues, before we lock in on the version
  - BobG: not sure that visibility is our goal for 3.0
  - Tim: have to figure out what matters to the TC. Vendors moving to 2.40 would be a big splash. Would be good to pull together informatkon about versions different vendors are using.
  - Valerie: also an issue of conformance, about what funcitonality different vendors support.

Motion to Adjourn

- Tim moves, Chris seconds. Motion passes and meeting adjourns at 15:11