From: Liu, Yi-Kai (Fed)
To: (b) (6)
Subject: Fw: Update to protocol integration.
Date: Saturday, June 25, 2016 1:37:17 AM

From: Emanuel Knill <knill@boulder.nist.gov>
Sent: Monday, June 13, 2016 5:53 PM
To: Bierhorst, Peter L. (Assoc)
Cc: Glancy, Scott (Fed); Liu, Yi-Kai (Fed); Jordan, Stephen P (Fed)
Subject: Update to protocol integration.

Here's an update to my protocol integration notes. As Peter noted, I had forgotten to include the "PASS/FAIL" events, and I added (at the bottom) a pass at an argument to go from Peter's source protocol promise to the one that I used to pass to the extractor.
See what you think.

Manny

%%%

The following is a specification for our algorithm combining a min-entropy source and a strong randomness extractor to produce a close-to-uniform string of bits according to the algorithm input, which can be considered as a request for a random string of bits. The point is to implement this algorithm top down, relying on existing extractor (Trevisan) and source analysis (Peter) protocols for the details. Notes discussing various aspects of the specification follow the specification.

Top level specification:

Input:
1. $1-\delta$, the level of the request--this is the security parameter.
2. σ, the number of bits requested.
3. ϵ', the requested nominal probability of success (the completeness parameter).

Output, provided the input request can be satisfied:
1. The output consists of strings M, S where M has σ bits and S is the string of all random bits used from non-quantum sources.
2. For any system described by state E satisfying the protocol assumptions, there exists a random variable U with values strings of σ bits, where U is uniformly distributed conditional on SE and there is an event G of (unconditional) probability at least $1-\delta$ such that $\Prob(U=M \vee \not \PASS| G)=1$.
3. While the success probability ($\Prob(\PASS)$) cannot be guaranteed to be at least ϵ', the presumption is that this was the design goal in the absence of major disruptions.

Informal protocol assumptions, alternative A (weaker physical, stronger E assumption, S is still private, "randomness expansion"):

1. Provided the internal physical (a) and random source constraints (b) are satisfied, E contains no information that could not already have been known before the algorithm was invoked.

Informal protocol assumptions, alternative B (stronger physical, weaker E assumptions, S may be public, "randomness refreshment"):

1. Provided the internal physical (c) constraints are satisfied And E effectively contains no information that could not already have been known before the algorithm was invoked, except for S, but where S satisfies random source constraints with respect to other initial information effectively known to E.

The top level specification is implemented by composing a quantum min-entropy source with an extractor. The min-entropy source may output $\PASS=0$, in which case the extractor does not need to be run and the failure propagates to the top level.

Extractor specification:

Input:

1. $1-\delta$, the level of the request.
2. C, a bit string.
3. δ_s, the smoothing parameter.
4. σ_s, the quantified min-entropy of C (see promise).
5. σ, the number of output bits requested.
6. Input promise: For any system described by state E_s satisfying source protocol assumptions, there exists a random variable V and an event G_s of probability at least $1-\delta_s$ such that V has maximum probability $2^{-\sigma_s}$ conditional on E_s and $\Prob(C=V |G_s)=1$. (For this algorithm, the source protocol assumptions are an implicit argument that propagates to the output.)

Output, provided the input request can be satisfied:

1. The output consists of M, S_x, where M has length σ and S_x is the random seed used.
2. Provided E_s satisfies extractor assumptions, there exists a random variable U with values strings of σ_x bits, where U is uniformly distributed conditional on $S_x E_s$ and there is an event G_x of (unconditional) probability at least $1-\delta$ Such that $\Prob(U=M|G_x)=1$.

Informal extractor assumptions, alternative A (seed is still private).

1. S_x is uniformly random given E_s.

Informal extractor assumptions, alternative B (seed is public).

1. E_s knows nothing "extra" beyond the value of S_x. I.e. $E_s = E_s(S_x,E'_s)$, where S_x is uniformly random given E'_s.

Source specification:

Input:

1. δ_s, the smoothing parameter, that is, the level of the request.

2. σ_s, the min-entropy requested.
3. ϵ', the requested nominal probability of success (the completeness parameter).

Output:

1. The output consists of C and S_s, the sequence of random bits used in choosing settings, and $\PASS$.
2. For any system with (unknown) current state E_s satisfying source assumptions, there exists an event G_s of probability at least $1-\delta_s$ and a random variable V such that the probability of V conditional on $S_s E_s$ is at most $2^{-\sigma_s}$ and $\Prob(V=C \vee \not \PASS|G_s)=1$.
3. See top-level output promise with regard to ϵ'.

Informal protocol assumptions, alternative A (weaker physical, stronger E_s assumption, S_s is still private):

1. Provided the internal physical (a) and random source constraints (b) are satisfied, E_s contains no information that could not already have been known before the algorithm was invoked.

Informal protocol assumptions, alternative B (stronger physical, weaker E_s assumptions, S_s may be public)

1. Provided the internal physical (c) constraints are satisfied And E_s effectively contains no information that could not already have been known before the algorithm was invoked, except for S_s, but where S_s satisfies random source constraints with respect to other initial information effectively known to E_s.

Parameter constraints.

Extractor parameters, according to Alan's slides and follow-ups:

Relevant input parameters: δ, δ_s, σ_s, σ, C.
Relevant algorithm parameters: s, the number of seed bits used.
And ϵ, the total variation distance to uniform of the output according to Alan's specification with a bit of slack to simplify.

Here are the extractor constraints:

1. $\epsilon+\delta_s \leq \delta$.
2. $\sigma\leq \sigma_s-4(\lceil\log_2(1/\epsilon)\rceil+6)$
3. $s=8(\lceil\log_2(4|C|/\epsilon^2)\rceil)^2$.

Source parameters, from Peter's notes.

Relevant source algorithm parameters, according to Peter's notes:
n, the number of trials, V, the "pass" statistic and
m, Bell function parameter (PR boxyness scale).
The number of settings choice bits used is $2n$.

Here is the source constraint:

1. $\delta_s \geq (1+2m(1-2^{-\sigma_s/n}))^n/V$
   See also [[Approximate%20min-entropy%20formula][Approximate min-entropy formula]].

Preprocessing:

We expect to be constrained by having n essentially fixed to get a reasonably high nominal probability of success. In preprocessing, we attempt to maximize σ subject to there being a solution to the constraints. Note that this may require additional information. With fixed n, the expected value of V can be anticipated and a conservative lower bound for the nominal probability of success can be used.

Notes:

+ The implementations of the specifications should be split into a pre-processing step that checks whether the input request can be satisfied and suggests alternative parameters if it cannot. This is one way of thinking about the process of determining goals in the context of source constraints.

+ I use the phrase "uniformly random given E" rather than uniform and independent of E because I am keeping in mind weakening this assumption, in which case independence will be lost but conditional distribution constraints still apply. It also means that the version of smoothed min-entropy used is defined parallel to the conditions on seeds.

+ Using probabilities of identity with "standard" RVs makes arguments on composing protocols simpler. I think the relationships between various "standard" distance measures and probability of identity with standard RVs is in one of the theses related to the field. Possibly Colbeck's. I am guessing that this type of promise plays a rule when studying composability of protocols. I can fill in relevant proofs later if needed. A relevant argument is below.

+ I moved ϵ' of Peter's notes to the top level, so in Peter's notes, δ_s from here is to be identified with $\delta\epsilon'$ from there. I did this because this parameter is rather "fuzzy", in the sense that there is no guarantee on what it is. At the same time, it doesn't play a fundamental role in the promises made by the different components: The promises are naturally unconditional. As a result, ϵ' is always just a multiplicative adjustment (deferrable to the top level) to allow for conditioning on success. All we can do is design for a nominal probability of success, usually close to 1 (as has always been the plan). Making a conservative choice of ϵ' to allow for a conditional-on-success promise is then up to the top-level user/caller/operator, if they want to. In a full protocol with period invocations of the algorithm, frequent success is essential for usability, and is sufficient support for not having to worry about probability of success (much). In practice, if we find even a few failures, there is likely an equipment problem requiring attention.

  I have been favoring the alternative of filling in with seed randomness in the presumably rare cases of failure. This disallows the alternative promise where the seeds are "public". I would note that for an application such as the beacon, we cannot fail to produce new randomness every minute (or less), so there always has

to be a backup plan, lest the beacon go out of service.
It is clear that this is a serious concern for Rene P.

+ <<Approximate min-entropy formula>> I expect that in our applications, $2^{-\sigma_s/n}$ is very close to
1; σ_s/n is the expected entropy rate, in bits per trial. The constraint for the source can then be approximated: $(1-2^{-\sigma_s/n}) = (1-e^{-\ln(2)\sigma_s/n}) \leq \ln(2)\sigma_s/n$. So the constraint is strengthened with $\delta_s \geq (1+2\ln(2)\sigma_{s}m/n)^n/V $. Using $(1+\alpha/n)^n \leq e^{\alpha}$, one can go for

\[\delta_s \geq 2^{2\sigma_{s}m}/V .\]

These approximations should be very good in our regime. The approximate expression gives some insight into the relationships between parameters. Observe that V is a product of individual trial statistics that is expected to grow exponentially. The expected growth is correlated with m, i.e. larger m means faster growth. A source for which there is a statistic with larger m should do better overall, after this relationship is accounted for.

+ In the top-level specification, we cannot promise $\Prob(U=M|SE)\geq1-\delta$ uniformly in SE. But we could promise $\Prob(\Prob(U=M|SE)\geq1-\delta')\geq1-\delta"$, where the outer probability is with respect to SE. Besides the complication of having an extra parameter, the relationship to the given promise is not direct: From the promise we can use a Markov bound to determine what can be promised in this conditional form. That is, from the promise we can conclude that $\sum_{se}\Prob(SE)\Prob(U\ne M|SE)\leq\delta$. So $\Prob(\Prob(U\ne M|SE)\geq\delta')\leq\delta/\delta'$.

+ It is necessary to have formal counterparts to the informal promises given. In stochastic process terms, they come down to the assertion that the state E of the system with respect to which we want the random bits produced to be uniform can be "sliced" into a stochastic process (E_k), where E is a function of (E_k) and (E_k) satisfies per-trial and pre-extractor conditions such as conditional uniformity of the random bits to be used as settings or seeds and no-signaling, as well as conditional independence statements that ensure that nothing is subsequently learned about the measurement outcomes. (That E can be made explicit in this way should follow from our proofs and is helpful when discussing the different incomparable security scenarios.) The slicing need not be the same as the physical time-slicing of the system. In particular, the random settings/seeds could have been learned earlier, provided that this didn't help, that is, it was known, but irrelevant as encoded in the virtual slicing (E_k). However, reality matters when justifying the assumptions on physical grounds. That is, if the settings/seeds were indeed public, we need to make extra physical assumptions about our devices, most notably the source. The "randomness refreshment" scenario is sensible if we built the source ourselves. If we bought it from QuantiQ and in the purchase agreed to licensing rules (proprietary innards!), this is harder to justify. We may have to trust independent physical barriers we put

in place.

If the "public input randomness" scenario is to be made explicit, we should have a name for it that parallels "randomness expansion" and "randomness amplification". One thought is that randomness that is (no longer) private and already known to all is stale, so the protocol refreshes stale randomness, so "randomness refreshment"? Only problem is it makes me think of a party with refreshments...

+ We need to pay attention to the issue of composability of source and extractor. I am not sufficiently familiar with the results on composability, I just know that it is a non-trivial issue. With the formulation of the promises given and all relevant systems being classical, this seems to be ok for now. Nevertheless, we should go over this explicitly.

+ As things stand, we have only made arguments for classical systems. This seems reasonable for now, particularly if we have built the source ourselves and/or are sure that the source is physically isolated except for the quantum channel required for the protocol. However, there are good reasons to consider quantum security seriously here. For example, we have one explicitly quantum device in the picture, and if we bought it from QuantiQ and have agreed not to or have no inclination to look inside, there is always the possibility that it takes full advantage of quantum mechanics to maximize its influence on the future. In particular, it can defer measurements of its own state, choosing them to optimize its correlation with measurement outcomes after having learned settings etc, for instance. Our current arguments do not apply to this possibility. I note that this issue has tripped up many researchers. Most notably Yao and his original quantum bit commitment scheme.

*** TV versus probability of identity.

Here's a lemma that helps with relating total variation distance to probability of identity with an RV having the target distribution:

Let μ be the probability distributions of RVs M and let μ' be another probability distribution on the range R of M. Suppose that the total variation distance between μ and μ' is δ. I'll take this to mean that $\int d\mu^+ =\delta$, where μ^+ is the positive part
of the signed measure $\mu-\mu'$. Claim: There exists a random variable U with the same range as M such that the probability distribution of U is μ' and $\Prob(M\ne U)=\delta$. Let μ^- be the positive part of
$\mu'-\mu$. Let E^+ and E^- be disjoint events such that $\mu^+(R\setminus
E^+)=\mu^-(R\setminus E^-)=0$. On E^+, $\mu'\le\mu$ and μ' is absolutely continuous with respect to μ, so the $d\mu'/d\mu\in[0,1]$ is defined. Similarly, on E^-, $\mu\leq\mu'$. Define B to be a RV with distribution independent of all relevant events so far. That is, form the independent product of the initial event space with another one on which B is defined, with probability distribution now being determined. Let B have range E^- and probability distribution

$(\mu'-\mu)/\delta$. Define a Markov process on M by the following
procedure:
If $m=M$, then: if $m\in E^+$, output $U=m$ with probability $d\mu'/d\mu$ and
$U=B$ with probability $1-d\mu'/d\mu$, otherwise output $U=m$. Then U
is distributed as μ' and $\Prob(U\ne M) = \delta$. The Markov process
must use new random resources, not part of the event space previously
in place. To formalize this better, we introduce a new independent
space/RV I with range $[0,1]$ and uniform probability
distribution. The Markov process can then be defined explicitly.
Note that the constructed RV is conditionally independent of
"everything else", given M.

Conversely, If there exists an RV U with distribution μ'
such that $\Prob(M\ne U)=\delta$, then the total variation distance
of the probability distribution of M from μ' is at most δ.

*** Smoothed min-entropy and how it relates to identity with min-entropy RV.

Here, I am assuming discrete, finite-range RVs. Otherwise this needs a
bit of work with conditional expectations and measurable sets up to
sets of measure zero and a better definition of V to avoid adding an
infinity of independent RVs. The construction bypasses the need
to explicitly refer to "smoothed min-entropy".

Peter's source protocol guarantees that $\Prob(\Prob(C|S_s E_s)\leq
2^{-\sigma_s} \vee \not\PASS)\geq 1-\delta_s$. $\Prob$ in the event
is evaluated with respect to the global, unconditional measure on the
total event space. From this we need to establish existence of the
random variable V in the source protocol output specifications. Note
that V can depend on the system with state E_s. We can assume that
the range of C has size N at least 2^{σ_s}. Fix $S_s=a,
$E_s = b$. Let (p_k) be the monotonically non-decreasing
probabilities of C given $S_s=a$ and $E_s=b$. Let k_0 be the
maximum k such that $p_k\leq 2^{-\sigma_s}$. Let $k(x)$ be the
position of the probability of x in the list (p_k). Let
$p_{>}=\sum_{k>k_0}p_k$. For $k\leq k_0$, define
$q_k=\beta(2^{-\sigma_s}-p_k)$ and for $k>k_0$, $q_k=\gamma
2^{-\sigma_s}$. Choose positive β and γ to satisfy
$\sum_k q_k = 1$, $\beta*p_{>}\leq 1$, and $\gamma*p_{>}\leq 1$. For
example, start with $\beta=\gamma=1/p_{>}$ to satisfy the latter two
constraints with $\sum_k q_k \ge 1$ (check that this is so
explicitly), then normalize. Let $A_{a,b}$ be a random variable with
probability distribution q_k independent of everything else. This
requires extending the global event space with a "hidden" independent
variable, one for each possible a and b. Define $V=C$ on the
event $[\Prob(C|S_s E_s)\leq 2^{-\sigma_s}]$. On the complement of
this event, define $V=A_{S_s,E_s}$. Then, conditionally on S_s and
E_s, V has min-entropy σ_s. It satisfies the condition on
V required for the output of the source protocol, with $G_s =
[\Prob(C|S_s E_s)\leq 2^{-\sigma_s} \vee \not\PASS]$.

Check of constraints for q_k: Let $\beta=\gamma=1/p_{>}$. Let
N be the size of the range of C and $n=2^{\sigma_s}$. Then
\begin{eqnarray}
\sum_k q_k &=& \sum_{k\leq k_0} \beta (1/n-p_k) + \sum_{k>k_0} \gamma/n \\
&=& (1/p_>)((k_0/n-(1-p_>)) + (N-k_0)/n) \\
&=& (1/p_>)(N/n-(1-p_>)) \\

```
  &\ge& (1/p_>)(n/n-(1-p_>)) = 1.
\end{eqnarray}

%%%
```