**Problem**

An application may be affected if, within 100 milliseconds, it performs

the following steps (which may be summarized as "read-fork-read-read"):

1. Read from the Crypto.Random PRNG, causing an internal reseed;

2. Fork the process and invoke Crypto.Random.atfork() in the child;

3. Read from the Crypto.Random PRNG again, in at least two different

processes (parent and child, or multiple children).

Only applications that invoke Crypto.Random.atfork() and perform the

above steps are affected by this issue. Other applications are

unaffected.

**Fix**

This is not part of the standard Fortuna definition and **using this function frequently** can weaken Fortuna's ability to **resist a state compromise extension attack**, but we **need this** **in order to properly implement Crypto.Random.atfork()**. Otherwise, forked child processes might continue to use their parent's PRNG state for up to 100ms in some cases. (e.g., CVE-2013-1445)

**FortunaAccumulator.py**

**def \_forget\_last\_reseed(self):**

**self.last\_reseed = None**

**\_UserFriendlyRNG.py**

**elf.\_fa.\_forget\_last\_reseed()**