FlipIt: A Game-Theory Handle on Password Reset and Other Renewal Defenses

Ari Juels

RSA, The Security Division of EMC

- Slides drawn from presentation by Prof. Ronald L. Rivest, MIT

Session ID: IAM-108

Session Classification: Intermediate

RSA CONFERENCE EUROPE 2012

Outline

- Overview and Context
- The Game of "FLIPIT"
- Non-Adaptive Play
- Adaptive Play
- Applications of FLIPIT
- Lessons
- Discussion

Cryptography

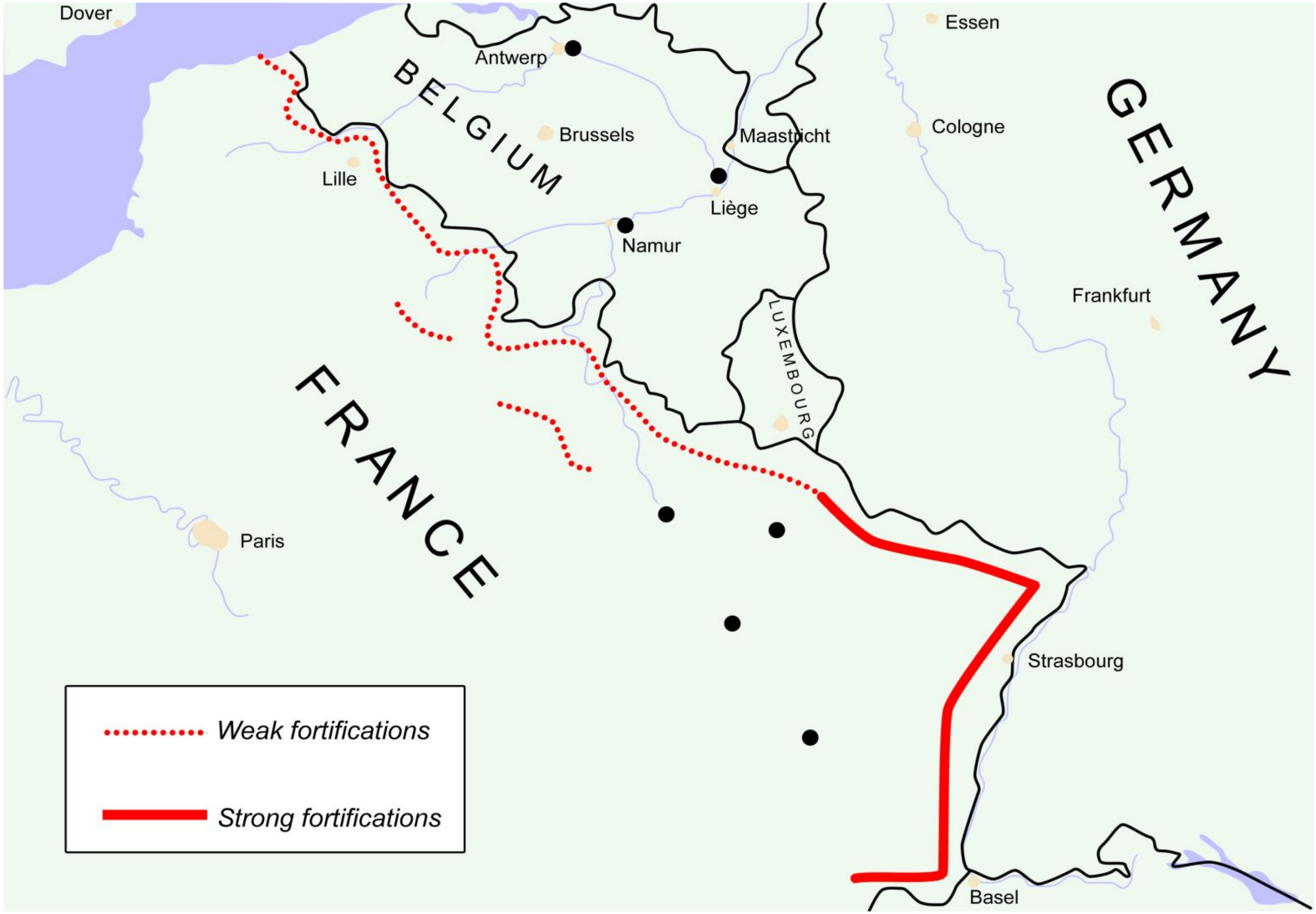
Cryptography is mostly about using *mathematics* and *secrets* to achieve confidentiality, integrity, or other security objectives.

Cryptography

We make assumptions as necessary, such as ability of parties to generate unpredictable keys and to keep them secret, or inability of adversary to perform certain computations.

Murphy's Law: "If anything can go wrong, it will!"

Assumptions may fail. Badly. (Maginot Line)



Even worse

In an adversarial situation, assumption may fail *repeatedly*...

(ref Advanced Persistent Threats)

Most crypto is like Maginot line...

We work hard to make up good keys and distribute them properly, then we sit back and wait for the attack.

There is a line we assume adversary can not cross (theft of keys).

Total key loss

To be a good security professional, there shouldn't be limits on your paranoia!

(The adversary won't respect such limits...)

Are we being sufficiently paranoid??

Lincoln's Riddle

Q: "If I call a dog's tail a leg, how many legs does it have?"

A: "Four. It doesn't matter what you *call* a tail; it is still a tail."

Corollary to Lincoln's Riddle

Calling a bit-string a "secret key" doesn't actually make it secret...

Rather, it just identifies it as an interesting target for the adversary!

Our goal

To develop new models for scenarios involving total key loss.

Especially those scenarios where theft is *stealthy* or *covert* (not immediately noticed by good guys).

To help develop a basic *science of cybersecurity*.

FlipIt

The Game of "FLIPIT"

(a.k.a. "Stealthy Takeover")

joint work with

Marten van Dijk, Alina Oprea, and Ronald L. Rivest

(RSA Labs & MIT)

FlipIt is a two-player game

- Defender = Player 0 = Blue
- Attacker = Player 1 = Red

FLIPIT is rather symmetric, and we say “player *i*” to refer to an arbitrary player.

There is a contested critical secret or resource

Examples:

- A password
- A digital signature key
- A computer system
- A mountain pass

State of secret or resource is binary

Good	Bad
Secret	Guessed or Stolen
Clean	Compromised
Controlled by Defender	Controlled by Attacker
Blue	Red

A player can “move” (take control) at any time

- Defender move puts resource into Good state

  = Initialize Reset Recover Disinfect

- Attacker move puts resource into Bad state

  = Compromise Corrupt Steal Infect

Time is *continuous*, not discrete.

Players move at same time with probability 0.

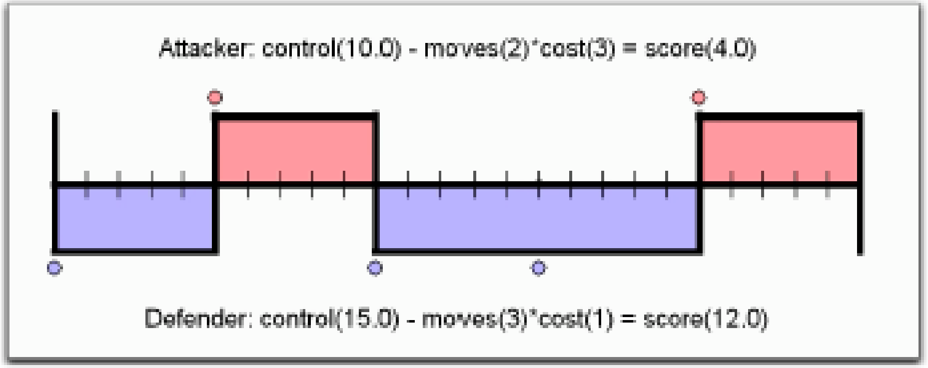
Examples of moves

- Create password or signing key
- Steal password or signing key

- Re-install system software.
- Use zero-day attack to install rootkit.

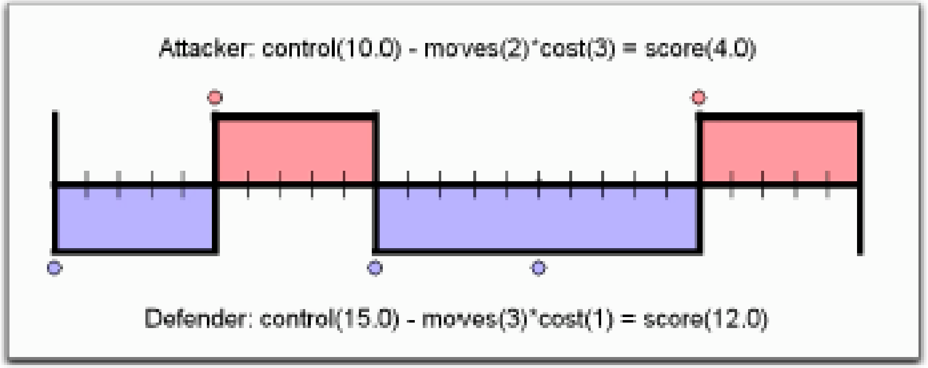
- Send soldiers to mountain pass.
- Send soldiers to mountain pass.

Continuous back-and-forth warfare…



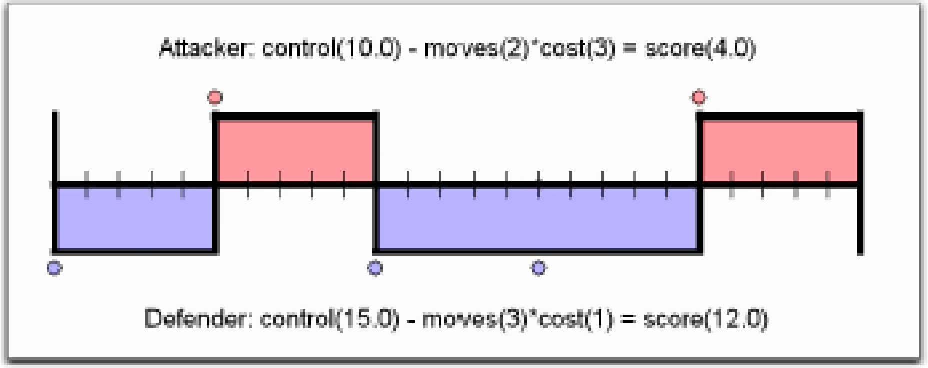
- Note that Attacker can take over at any time.
- There is no "perfect defense."
- Only option for Defender is to re-take control later by moving again.
- The game may go on forever…

Moves are "stealthy"



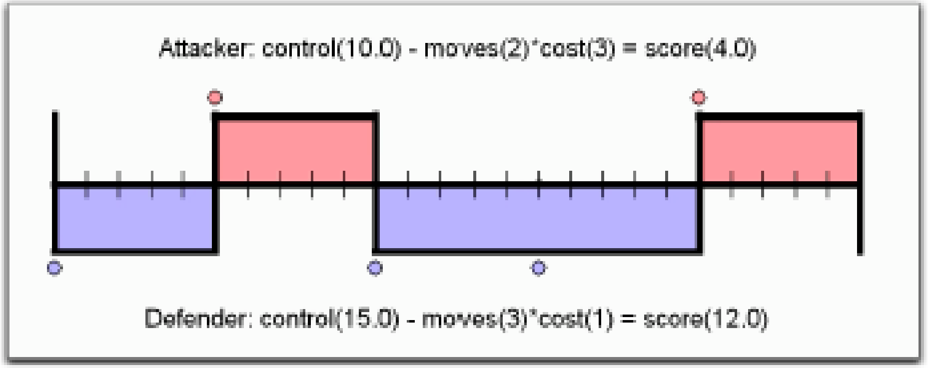
- In practice, compromise is often undetected...
- In FLIPIT, players do *not* immediately know when the other player makes a move! (Unusual in game theory literature!)
- Player's uncertainty about system state increases with time since his last move.

Moves are “stealthy”



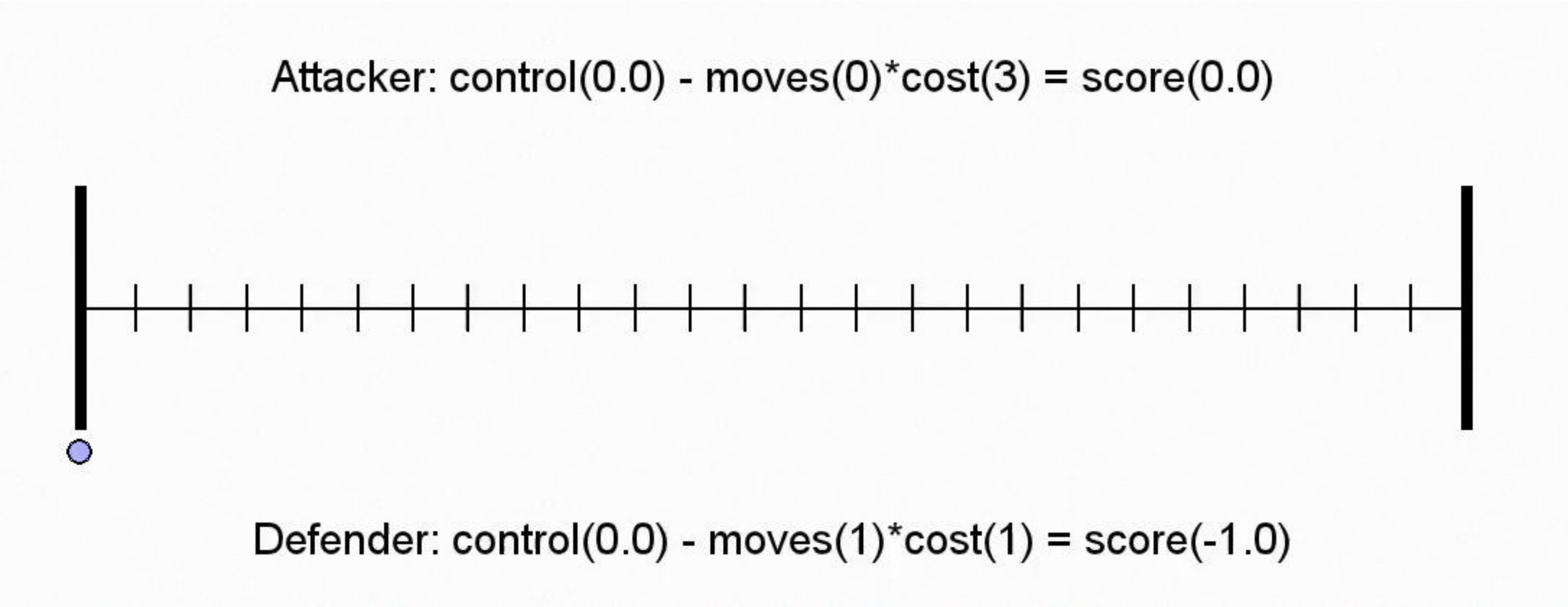
- A move may take control (“flip”) or have no effect (“flop”).
- Uncertainty means flops are unavoidable.

Moves may be informative

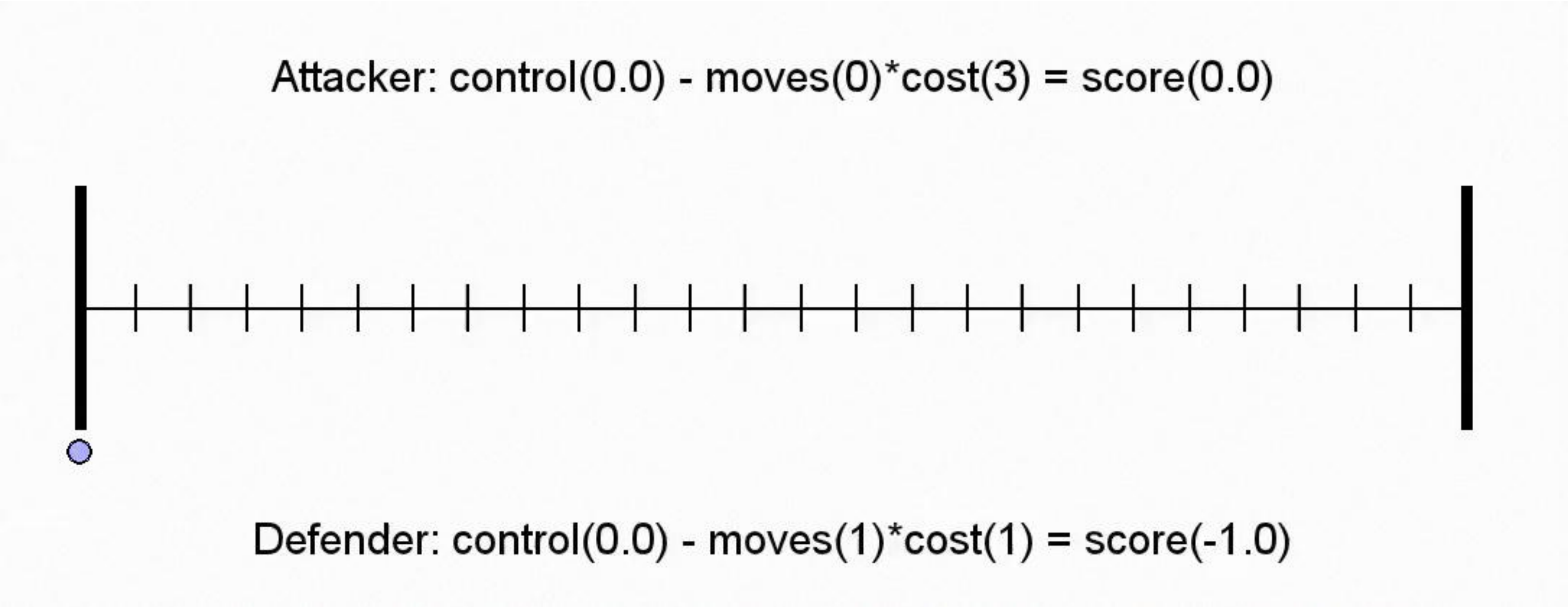


- A player learns the state of the system only when she moves.
- In basic `FLIPIT`, each move has feedback that reveals all previous moves.
- (In variants, move reveals only current state, or time since other player last moved...)

Movie of `FLIPIT` game, global view



Movie of FLIPIT game, defender's view



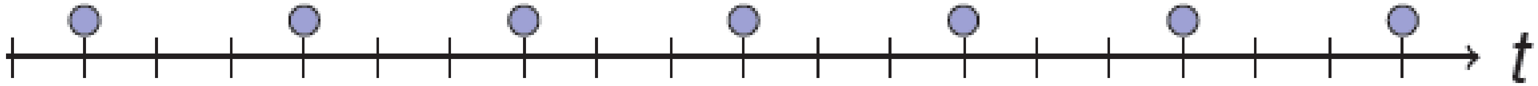
How to play `FlipIt` well?

Non-adaptive play

- A non-adaptive strategy plays on blindly, independent of other player's moves.
- In principle, a non-adaptive player can pre-compute his entire (infinite!) list of moves before the game starts.
- Some interesting non-adaptive strategies:
  - Periodic play
  - Exponential (memoryless) play
  - Renewal strategies: i.i.d. intermove times

Periodic play

- Player *i* may play *periodically* with rate α_i and period $1/\alpha_i$
  - E.g. for $\alpha_0 = 1/3$, we might have:



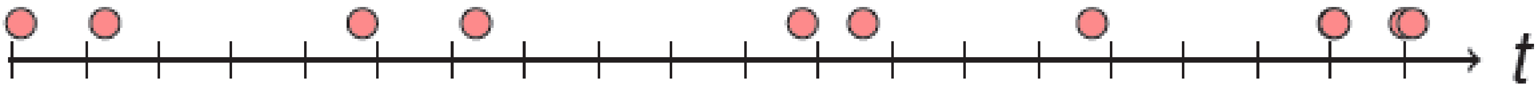
Exponential play

If Attacker plays exponentially with rate α_1, then her moves form a memoryless Poisson process; she plays independently in each interval of time of size dt with probability α_1 dt.

Probability that intermove delay is at most *x* is

$$1 - e^{-\alpha_1 x}$$

E.g., for $\alpha_1 = 1/2$, we might have:



Non-adaptive play

A key theorem: Among a large class of non-adaptive strategies (renewal strategies) for Attacker and Defender, the optimal strategy is either periodic or not playing at all.

Adaptive play

- An *adaptive* player pays attention to her opponent's moves and adjusts her play accordingly.
- Periodic strategy not very effective against adaptive Attacker, who can learn to move just after each Defender move.
- Examining periodic vs. adaptive play yields our first, simple lesson: Standard password reset policies are badly conceived!

Password reset

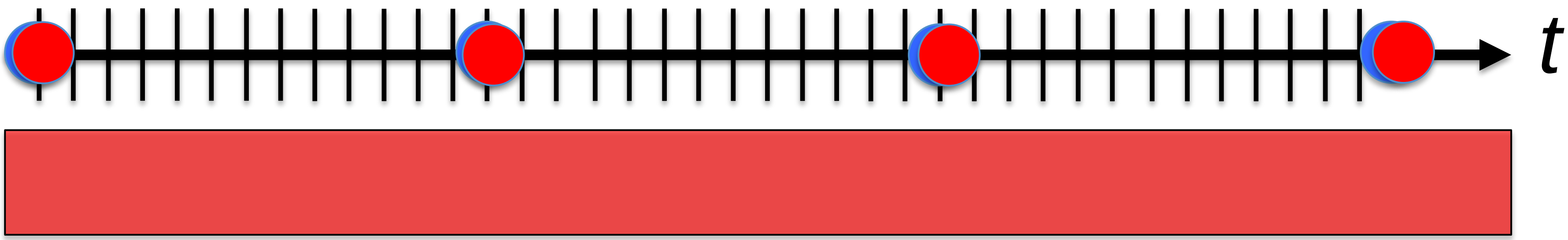
- Password reset can be modeled in `FLIPIT`
  - The Defender takes control by resetting his password.
  - The Attacker takes control by stealing a password.
- Both actions have an associated cost
  - Passwords can be purchased online in underground markets; tens of dollars for a consumer e-mail password
  - Password reset has a human cost; help-desk costs for password reset suggest a cost of tens of dollars.

Password reset

- A Defender benefits by controlling her e-mail account: Her identity is not subject to misuse.
- An Attacker benefits by controlling a stolen e-mail account: It may be used to send spam, facilitate identity theft, etc.
- Most organizations require users to reset their passwords at regular intervals, e.g., every 90 days.

Standard password reset:

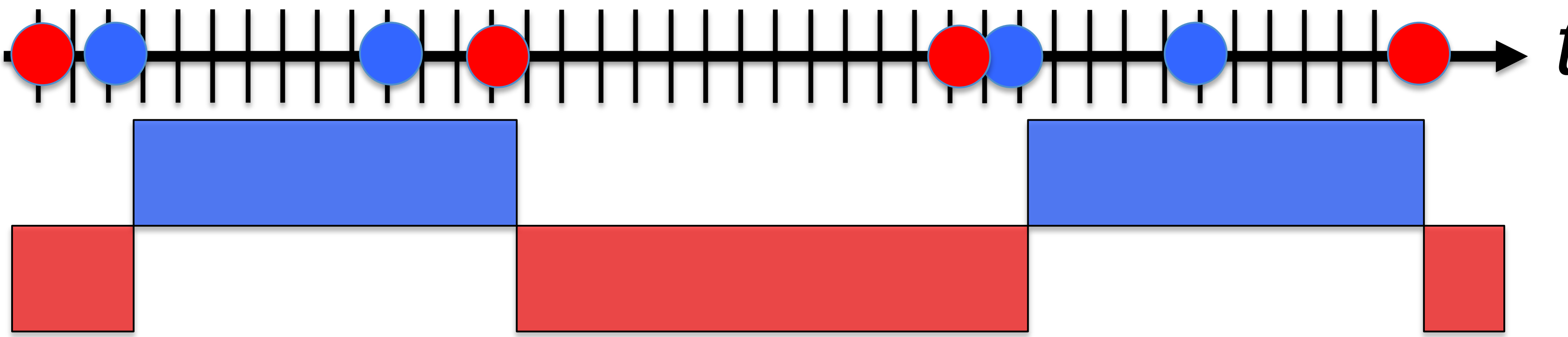
- periodic (90-day interval) vs.
- adaptive



Can we do better?

Alternative password reset

exponential (90-day mean) vs.
adaptive

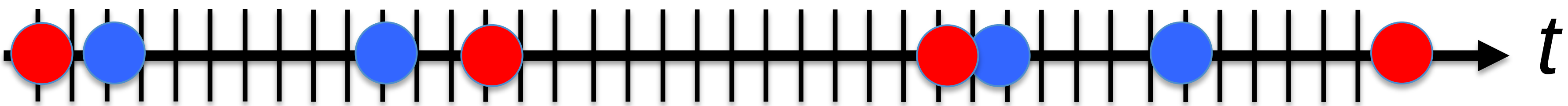


- For realistic parameterizations, Attacker will control resource a majority of the time
- But Defender will have *much more control* than with periodic password reset

Optimal password reset

???? vs.

adaptive

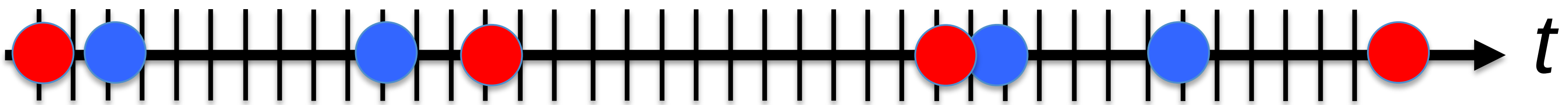


- We do know that we can do slightly better than exponential
- *Delayed Exponential* (*DE*): Wait *X* days, and then move exponentially
- Also ensures users aren't hit with immediate, sequential resets

Choosing parameters

???? vs.

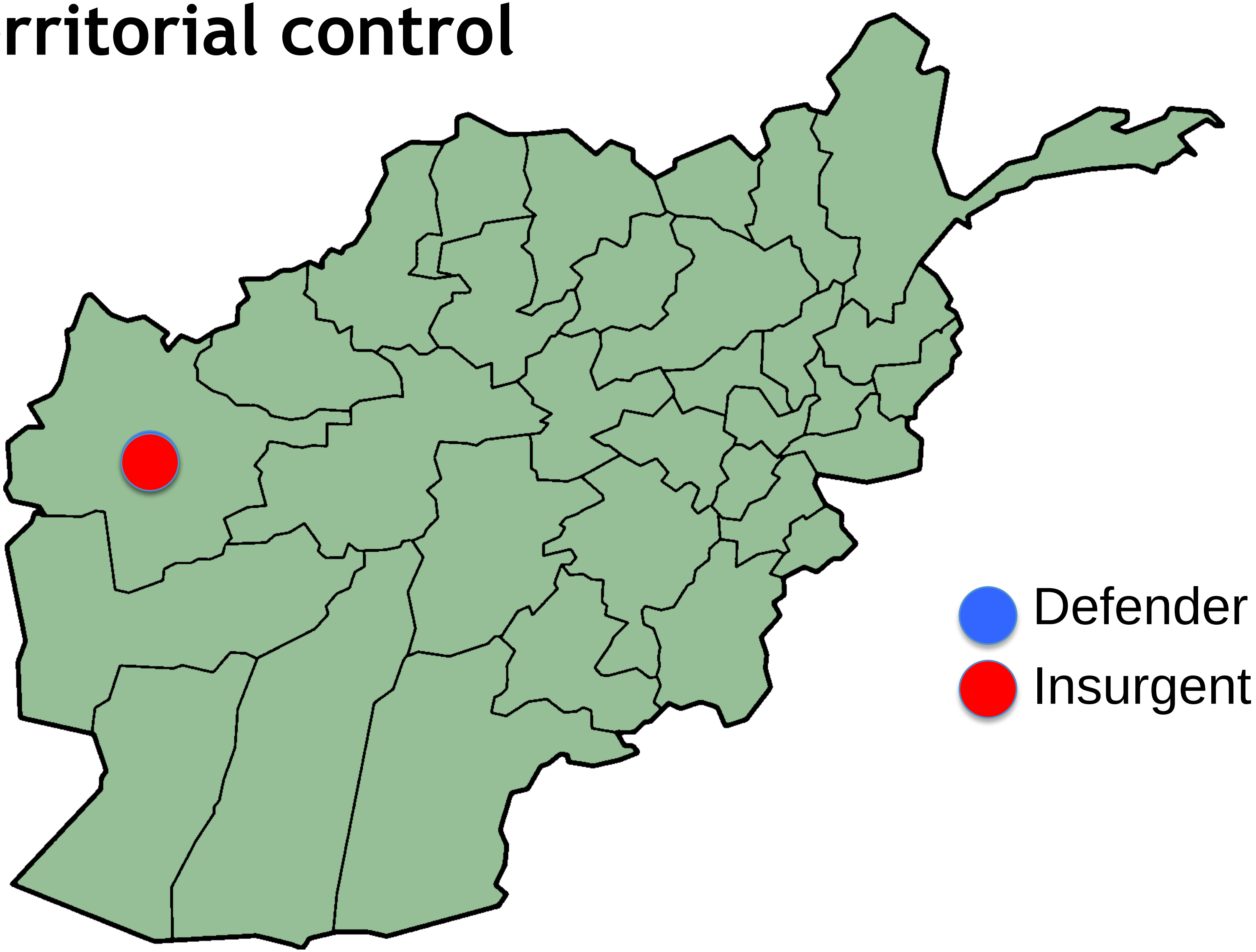
adaptive



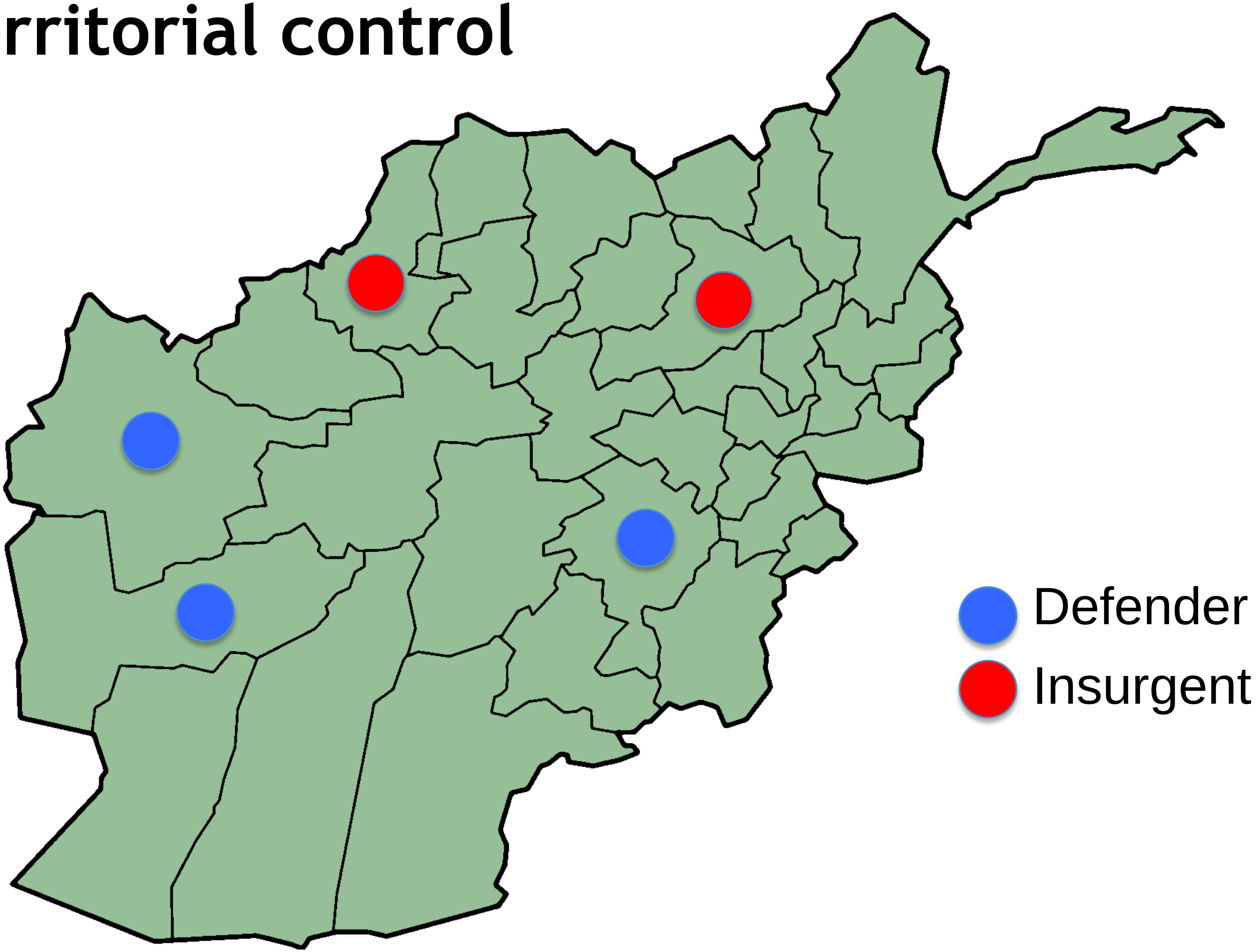
- We can estimate costs as already suggested (e.g., costs of help-desk calls)
- But the best approach is probably just to choose something “reasonable,” e.g.,
  - *X* = 10 days; DE mean = 90 days

Where else might `FlipIt` be applied?

Territorial control



Territorial control



Territorial control (cyberspace)

Audit of treaties / cloud security

- Not enriching uranium
- Enriching uranium

- Encrypting files at rest
- Not encrypting files at rest

Lessons

1. Be prepared to deal with repeated total failure (loss of control).
2. Play fast! Aim to make opponent drop out (Agility!)
   - (Reboot server frequently; change password often)
3. Arrange game so that your moves cost much less than your opponent's!
   - Cheap to refresh passwords or keys, easy to reset system to pristine state (as with a virtual machine)

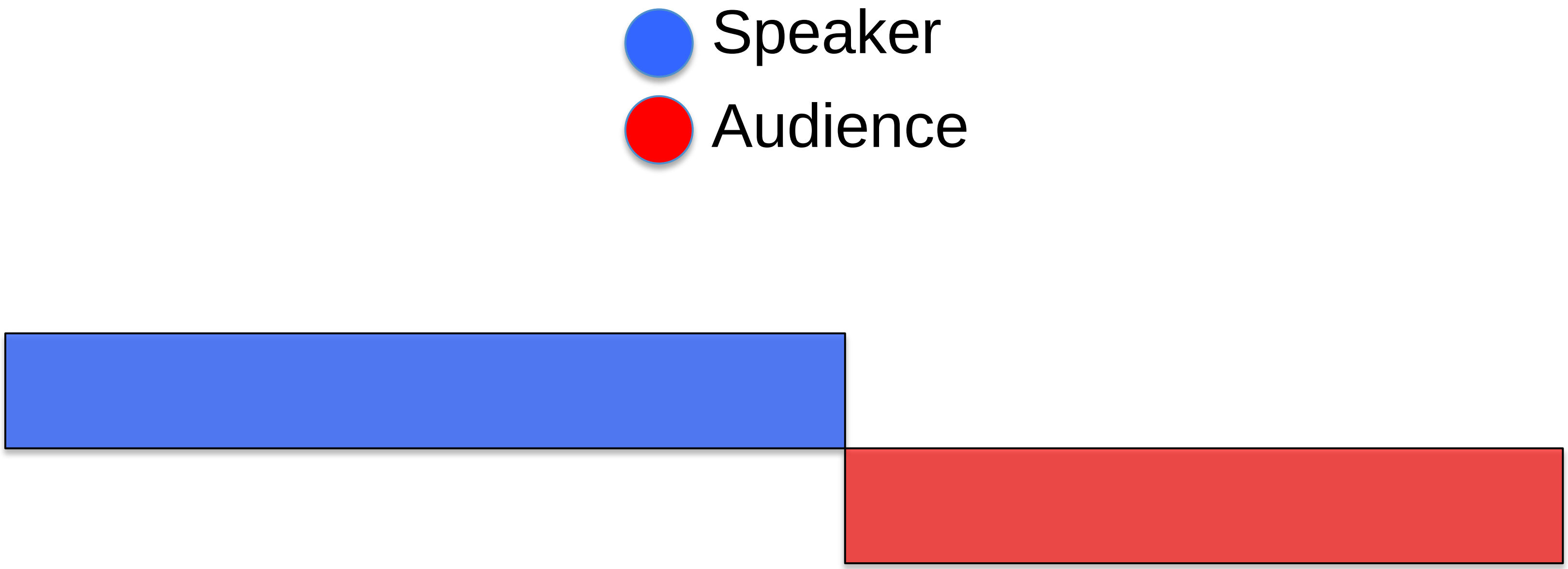
Apply Slide

- If you read about `FlipIt`, you'll probably find applications we haven't thought of

In any case, you might…

- Randomize your password reset intervals
- Design new infrastructure to be *agile*, i.e., low cost in the `FlipIt` sense
  - E.g., allow virtual machines to be easily rebuilt

Over to you…



Visit

`www.rsa.com/flipit`