Chameleon Hashes in the Forward-Secure ID-Based Setting

Madeline González Muñiz* and Peeter Laud

Theory Days
Tõrve, Estonia

October 8, 2011

CYBERNETICA

MOTIVATION FOR CHAMELEON HASHING

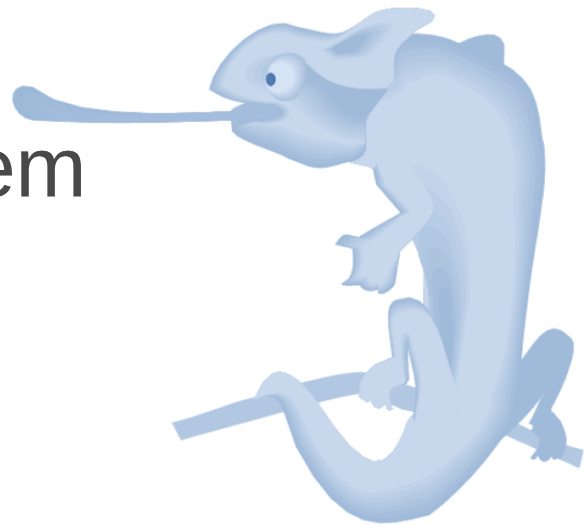
CYBERNETICA

Sanitizable Signature Schemes

- Allow modification to the original message
  - Pre-determined deletion
  - Pre-determined modification
    - ✓ Chameleon hashes
- Sender→Sanitizer→Receiver

Chameleon Hashes

- Introduced by Krawczyk and Rabin in 2000
- Collision-resistant with a trapdoor for finding collisions
- Key exposure problem
- Non-transferable



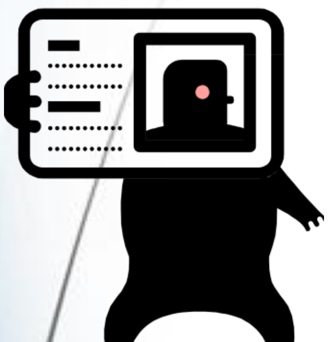
CYBERNETICA

Key Exposure Problem [KR2000]

- For public key $y=g^x \bmod p$
- Hash defined as $h(m, r)=g^m y^r \bmod p$
- One can solve for x given (m, r) and (m', r') such that $g^m y^r = g^{m'} y^{r'}$

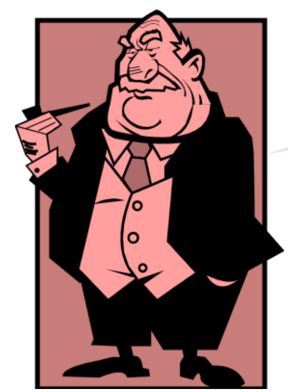
PRELIMINARIES

Identity-Based Cryptography

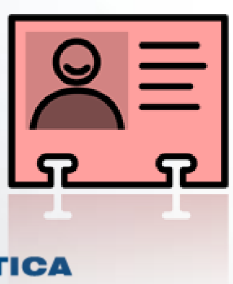


Authenticate to Key Generator

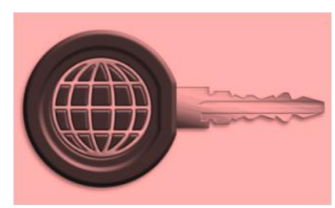
Key Generator gives ID a private key for the system



Has a master public/private key



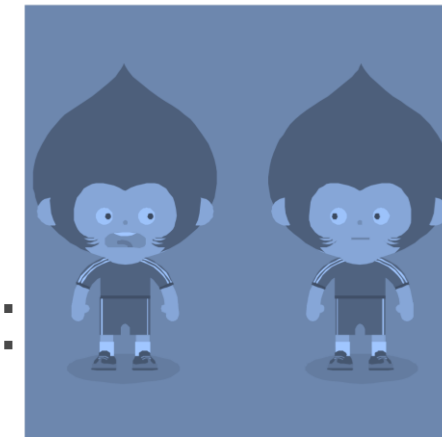
Public key computed from ID



CYBERNETICA

Bilinear Map (Pairing)

Let $\mathbb{G}_1$ (+) and $\mathbb{G}_2$ ($\cdot$) be two groups of prime order q



e: $\mathbb{G}_1$X $\mathbb{G}_1 \rightarrow \mathbb{G}_2$ a bilinear map:

1. Bilinear:
   $e(\alpha P, \beta Q) = e(P, Q)^{\alpha\beta}$
2. Non-degenerate
3. Efficiently computable

Bilinear Computational Diffie-Hellman Problem

Given P, αP, βP, γP, compute:

$$e(P, P)^{\alpha\beta\gamma}$$

We will refer to this as BCDH

CYBERNETICA

Bilinear Decisional Diffie-Hellman Problem

Given P, αP, βP, γP, decide:

random element in G_2 or $e(P, P)^{\alpha\beta\gamma}$

We will refer to this as BDDH

CYBERNETICA

Pseudorandom Bit Generator

- Bellare and Yee 2003
- G=(G_k, G_n, k, T)
  - ➢ G_k takes no input, outputs $Seed_0$
  - ➢ G_n deterministically takes input $Seed_{t\text{-}1}$, outputs (Out_t, $Seed_t$) where Out_t is a k-bit block and runs a max of T times
- Indistinguishable from a function that outputs k-bit blocks unif at random

CHAMELEON HASHES IN ID-BASED SETTING W/O KEY EXPOSURE

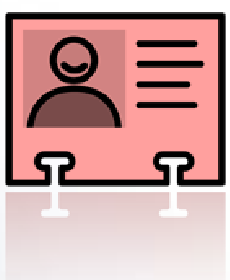
Chen et al. 2010 Proposed Scheme

- Setup

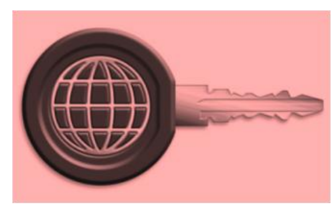
e: $G_1 X\, G_1 \rightarrow G_2$

Master Secret key s

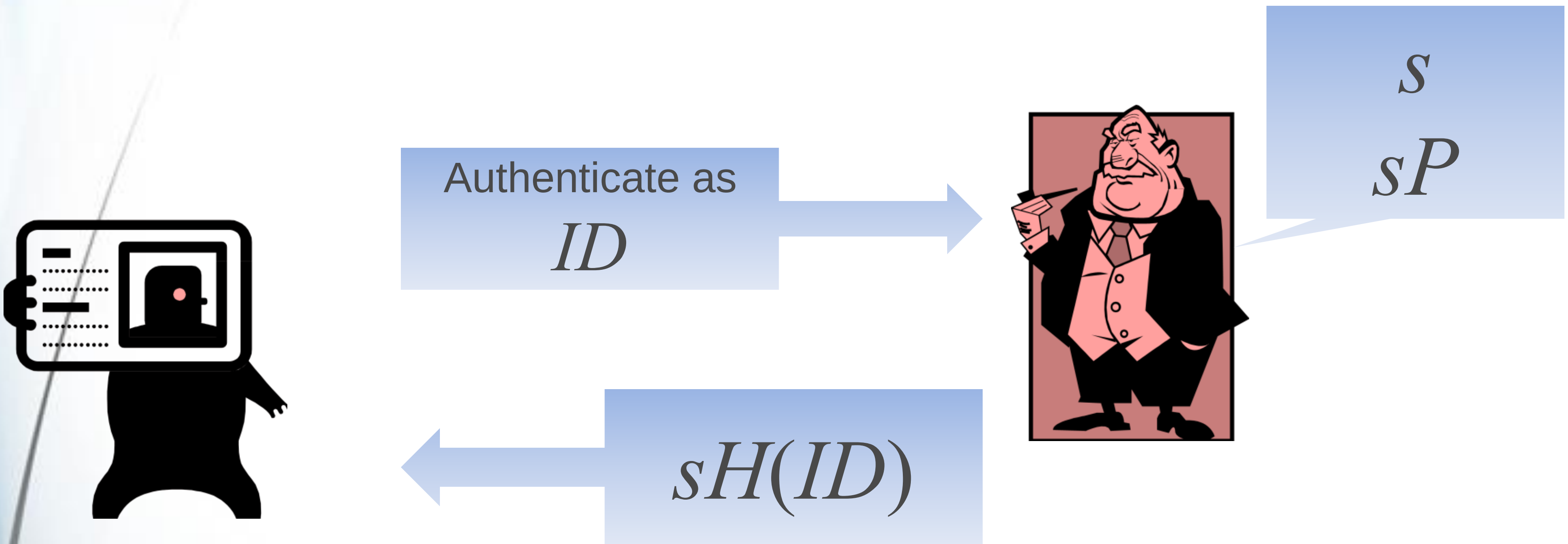
Master Public key sP



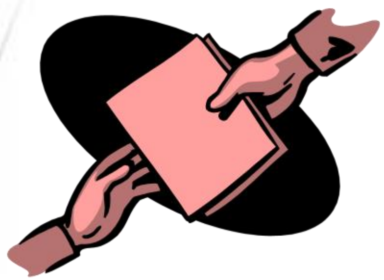
$H(ID)$



Key Extraction



Chameleon Hash



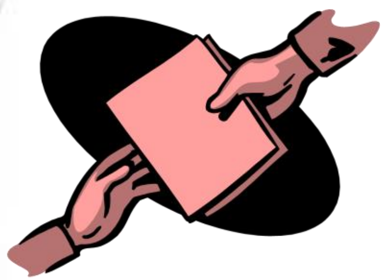
Sender

- Select a uniformly at random
- $r=(aP, e(a(sP), H(ID))$
- $h=aP+mH_1(L)$

public $H(ID)$

L is a transaction label

Collision (Forgery) by ID



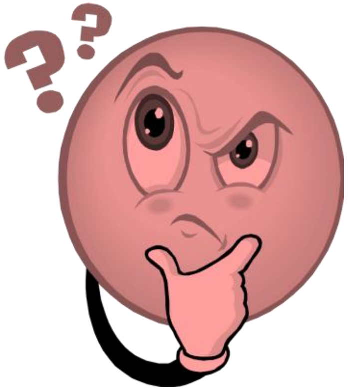
- Select message m'
- $a'P = aP + (m - m')\, H_1(L)$
- $r' = (a'P,\ e(a'P,\ sH(ID)))$

The proof relies on the difficulty of computing the second component of r'

private
$sH(ID)$

The Problem

- Who can verify the correctness of the second component of r and r'?
  - Sender knows discrete log a
  - Forger using private key
  - BDDH easy
- Solution
  - Include a NIZK proof



SECURITY MODEL W/ FORWARD SECURITY

CYBERNETICA

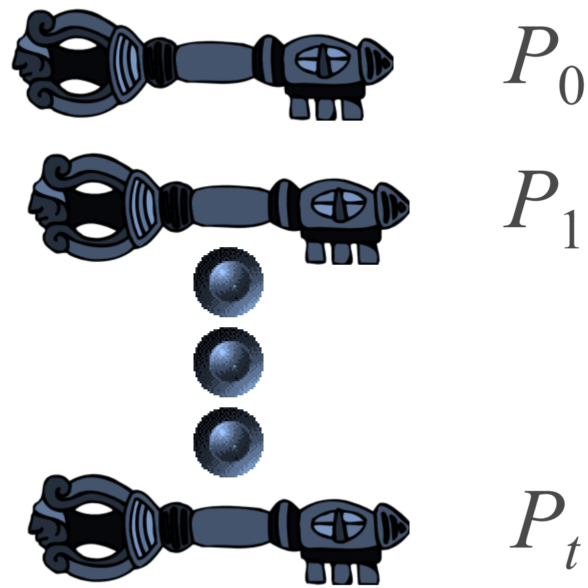
Properties

- Forward-secure collision resistance
- Indistinguishability

Forward-Secure Collision Resistance

» Users in the system are honest

$params$



SK_{ID} for break-in time t

Collision Forgery

» For $t' < t$

$P_{t'}, ID', L, m, r$

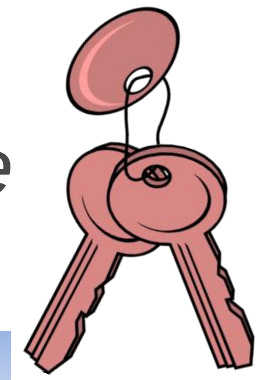
$P_{t'}, ID', L, m', r'$

Same hash output

Indistinguishability

$params$

Extraction Oracle



P_t, ID, L, m

$h(P_t, ID, L, m, r)$

$h(P_t, ID, L, m^*, r)$

PROPOSED CONSTRUCTION

Proposed Forward-Secure KGC Model

e: $G_1 X\ G_1 \rightarrow G_2$

$G=(G_k, G_n, k, T)$

At time $t=0$

Master secret key $S_0=(s_0, Seed_0)$

Master public key $P_0= s_0P$

Given $S_{t\text{-}1}=(s_{t\text{-}1}, Seed_{t\text{-}1})$

$G_n\ (Seed_{t\text{-}1})=(Out_t, Seed_t)$

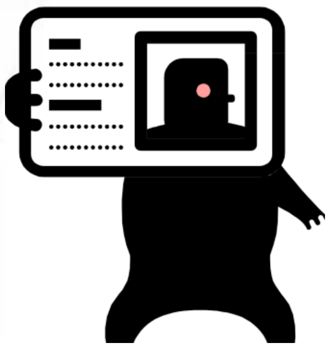
Compute $s_t= H(Out_t)s_{t\text{-}1}$

Master secret key $S_{\mathrm{t}}=(s_t, Seed_t)$

Master public key $P_t=s_tP$

Master Key Update

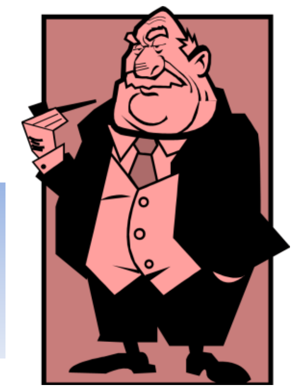
Key Extraction and Identity Update



Authenticate as

ID

$s_t H(ID), P_t$



Given $S_{t\text{-}1}=(s_{t\text{-}1}H(ID), Seed_{t\text{-}1}), P_{t\text{-}1}$

$G_n\ (Seed_{t\text{-}1})=(Out_t, Seed_t)$

User secret key $S_t=(H(Out_t)s_{t\text{-}1}H(ID), Seed_t)$

$=(s_t H(ID), Seed_t)$

Master public key $P_t= H(Out_t)P_{t\text{-}1}$

User Key Update

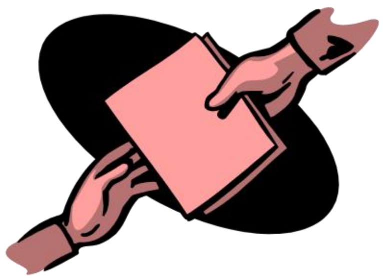
CYBERNETICA

Hashing Algorithm

Sender

- Select a uniformly at random
- $r=(aP, e(aP_t, H(ID)))$
- $h=aP+mH_1(L)$ and NIZK π that r was correctly formed

Collision (Forging) Algorithm



Receiver

- Select message m'
- $a'P = aP + (m - m')\, H_1(L)$
- $r' = (a'P,\ e(a'P,\ s_t H(ID)))$
- NIZK π' that r' was correctly formed

SECURITY OF PROPOSED CONSTRUCTION

BCDH Reduction

Challenger

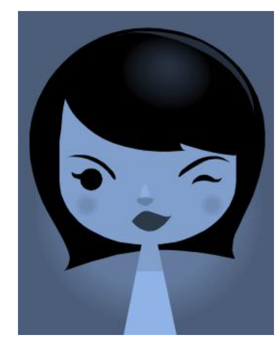
$P, \alpha P, \beta P, \gamma P$

$e(P, P)^{\alpha\beta\gamma}$

A can create a collision in the hash

B interacts with A to solve BCDH

B



A

Collision Resistance

- Assumption that BCDH is hard
- Using the second component of r and r' we have the following:
  - $e(a'P, s_t H(ID))$
    $= e(aP + (m\text{-}m')\, H_1(L), s_t H(ID))$
    $= e(aP, s_t H(ID))\; e(H_1(L), s_t H(ID))^{m\text{-}m'}$
  - $e(a'P, s_t H(ID))$ / $e(aP, s_t H(ID))$
    $= e(s_t H(ID), H_1(L))^{m\text{-}m'}$
  - $e(s_t H(ID), H_1(L))$ used in simulation to introduce challenge

CYBERNETICA

BCDH Challenge

Given P

$\alpha P = P_t = s_t P$

$\beta P = H(ID)$

$\gamma P = H_1(L)$

compute:

$$e(s_t H(ID), H_1(L)) = e(P, P)^{\alpha\beta\gamma}$$

Open Problem

- Attribute-based setting
  - User with threshold number of attributes can compute collision
  - Sahai and Waters
    - Public parameter for each attribute
  - Chameleon hash with the following condition:
    - Hash depends on message, attributes, and attribute authority's public key
    - User and attribute authority interact once

THANKS