Formal Analysis of Electronic Exams

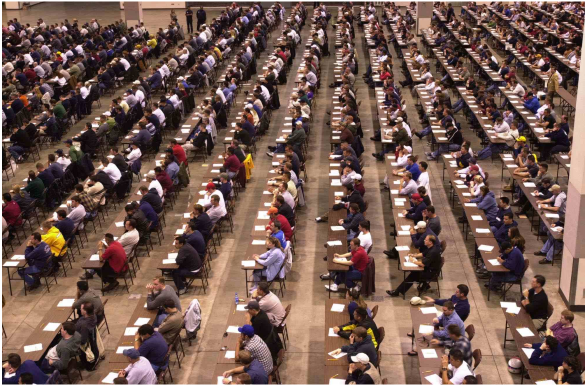
Jannik Dreier[1], Rosario Giustolisi[2], Ali Kassem[3], Pascal Lafourcade[4], Gabriele Lenzini[2] and Peter Y. A. Ryan[2]

[1]Institute of Information Security, ETH Zurich
[2]SnT/University of Luxembourg
[3]Université Grenoble Alpes, CNRS, VERIMAG
[4]University d'Auvergne, LIMOS

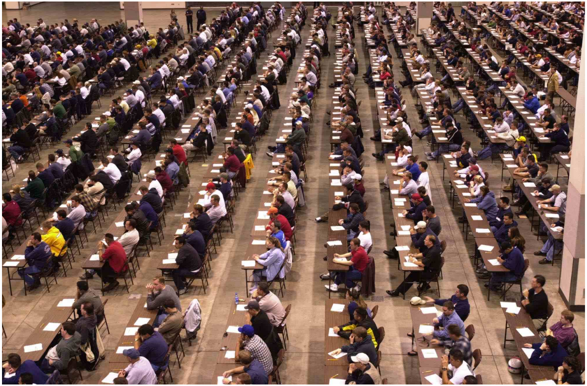
11th International Conference on Security and Cryptography (SECRYPT 2014), Vienna

August 28, 2014

E-exam



E-exam



Information technology for the assessment of knowledge and skills.

Educational assessment

coursera

UDACITY

edX

IELTS™
English for International Opportunity

CISCO™
CCNA

TOEFL iBT

E-exam: Players and Organization

Three Roles:

Candidate

Examination Authority

Examiner

E-exam: Players and Organization

Three Roles:

Candidate

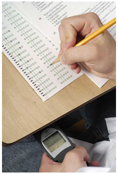
Examination Authority

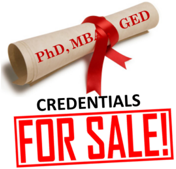
Examiner

Four Phases:

1. Registration 2. Examination 3. Marking 4. Notification

Threats. . .





- Candidate cheating
- Bribed, corrupted or unfair examiners
- Dishonest/untrusted exam authority
- Outside attackers
- . . .

. . . and their Mitigation

Most existing e-exam systems assume **trusted authorities** and focus on **student cheating**:

- Exam centers

- Software solutions, e.g. ProctorU

ProctorU
Real People.
Real Proctoring.

. . . and their Mitigation

Most existing e-exam systems assume **trusted authorities** and focus on **student cheating**:

- Exam centers

- Software solutions, e.g. ProctorU

Yet also the **other threats** are real:

- Atlanta Public Schools cheating scandal (2009)
- UK student visa tests fraud (2014)

. . . and their Mitigation

Most existing e-exam systems assume **trusted authorities** and focus on **student cheating**:

- Exam centers

- Software solutions, e.g. ProctorU

Yet also the **other threats** are real:

- Atlanta Public Schools cheating scandal (2009)
- UK student visa tests fraud (2014)

So what about **dishonest authorities** or **hackers** attacking the system?

. . . and their Mitigation

Most existing e-exam systems assume **trusted authorities** and focus on **student cheating**:

- Exam centers

- Software solutions, e.g. ProctorU

Yet also the **other threats** are real:

- Atlanta Public Schools cheating scandal (2009)
- UK student visa tests fraud (2014)

So what about **dishonest authorities** or **hackers** attacking the system?

$\Rightarrow$ need for better protocols and systems (cf. case studies)

. . . and their Mitigation

Most existing e-exam systems assume **trusted authorities** and focus on **student cheating**:

- Exam centers
- Software solutions, e.g. ProctorU

Yet also the **other threats** are real:

- Atlanta Public Schools cheating scandal (2009)
- UK student visa tests fraud (2014)

So what about **dishonest authorities** or **hackers** attacking the system?

⇒ need for better protocols and systems (cf. case studies)

⇒ **precise formal definitions of required properties**

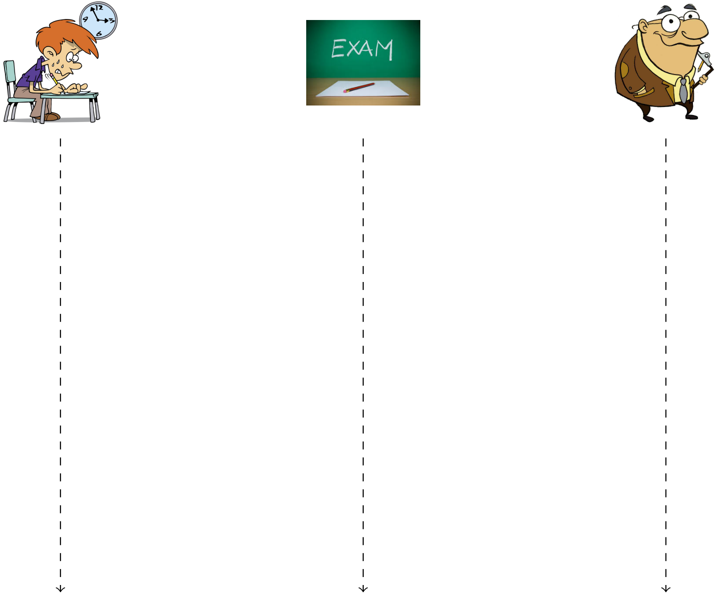
Plan

Plan

Model

- **Processes** in the applied π-calculus [?]
- Annotated using **events**
- **Authentication** properties as **correspondence** between events
- **Privacy** properties as **observational equivalence** between instances
- **Automatic** verification using ProVerif [?]

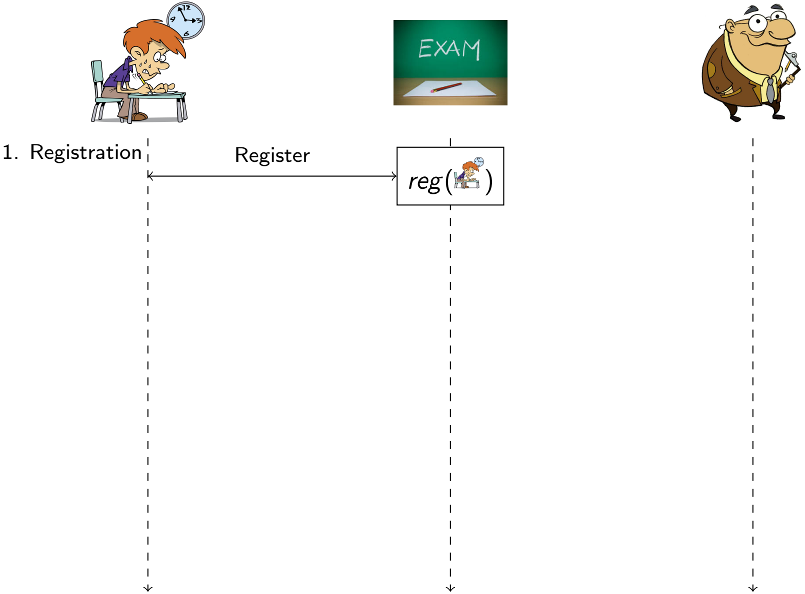
Model



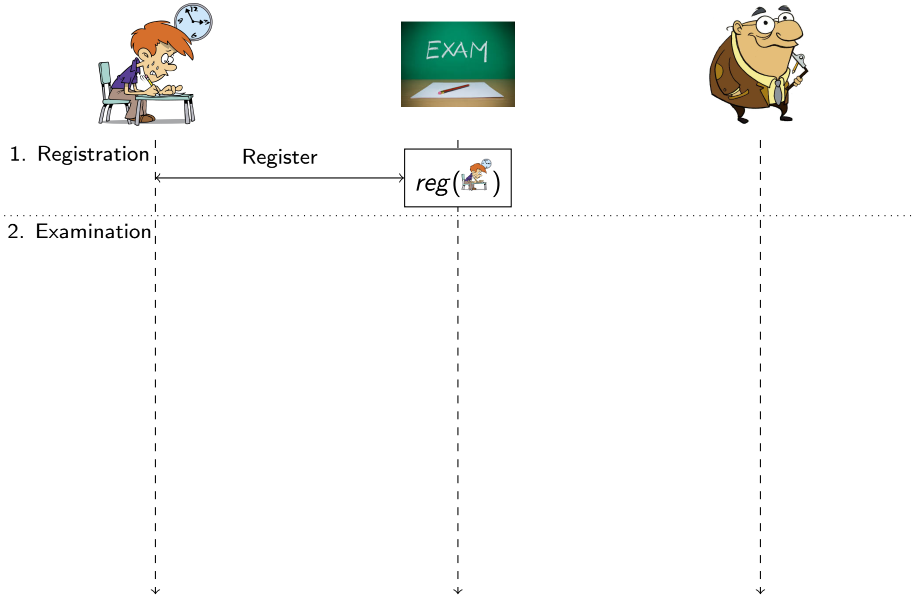
Model

1. Registration

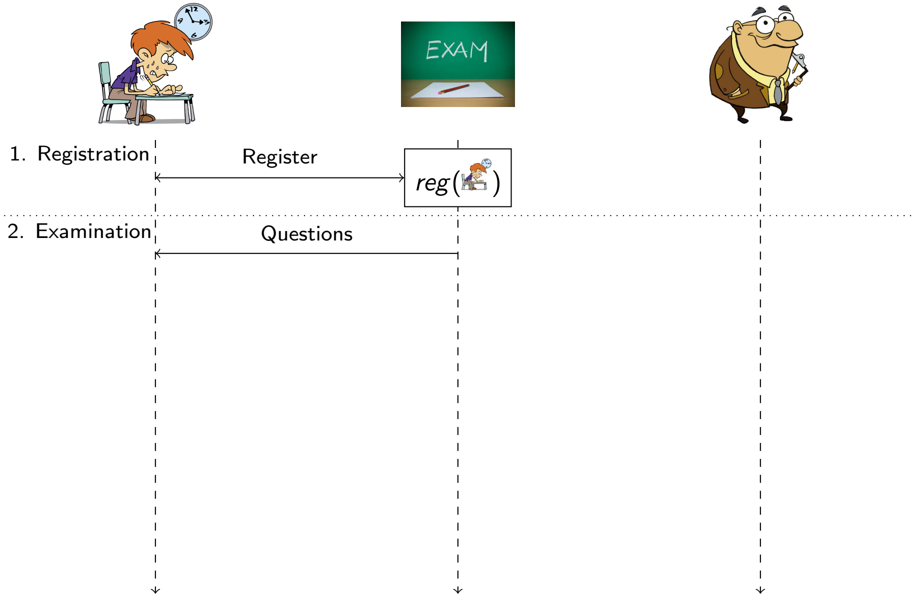
Model



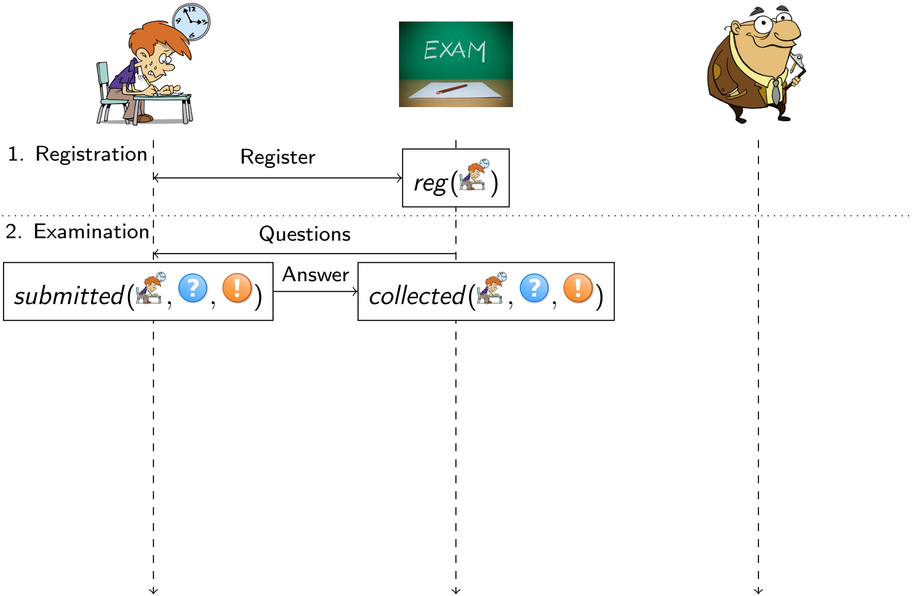
Model



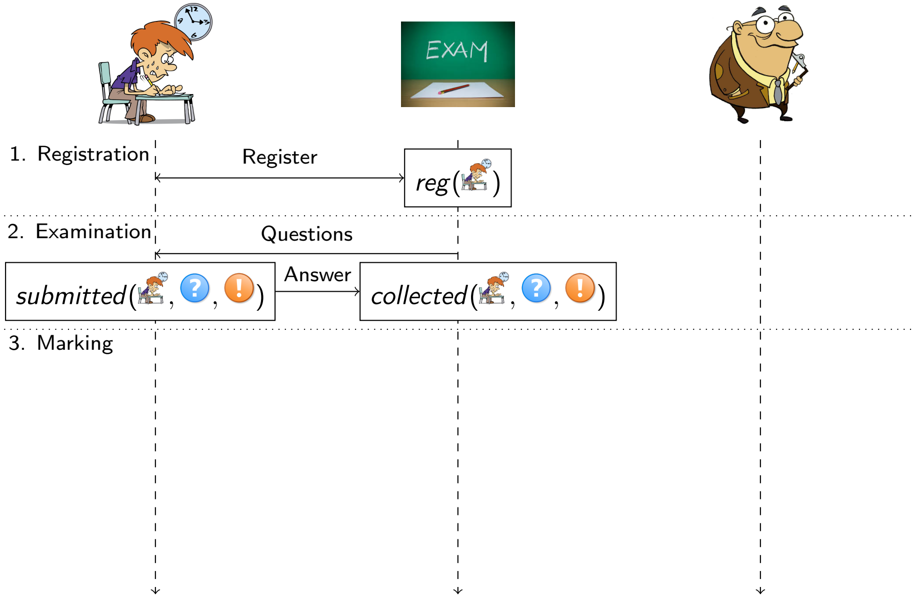
Model



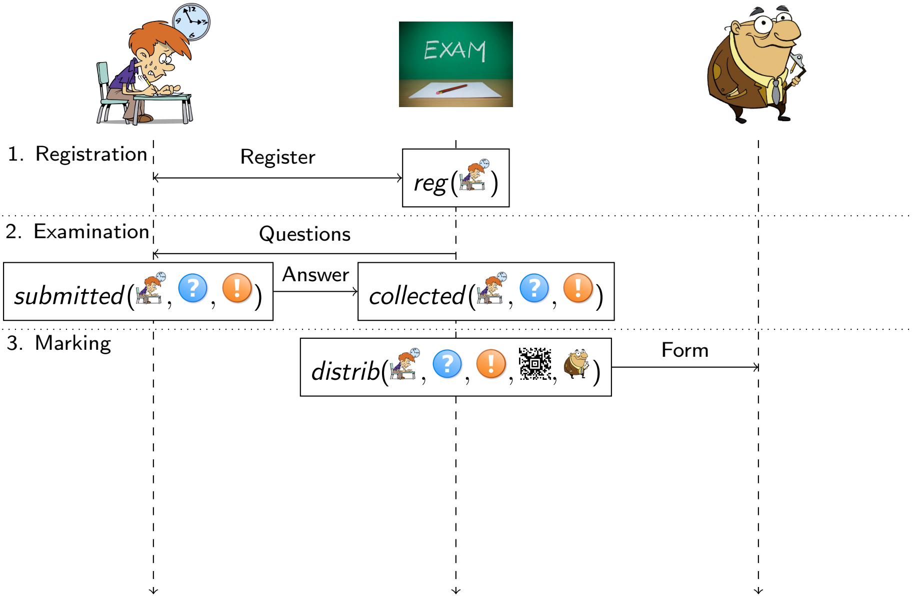
Model


EXAM
1. Registration
Register
reg( )
2. Examination
Questions
submitted(, ?, !)
Answer
collected(, ?, !)

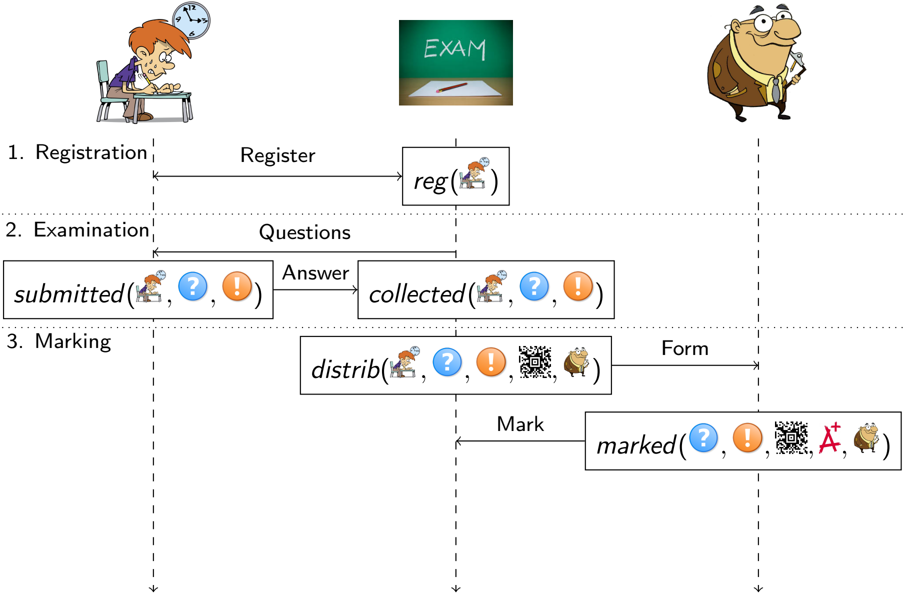
Model


EXAM
1. Registration
Register
reg( )
2. Examination
Questions
submitted(, ,)
Answer
collected(, ,)
3. Marking

Model

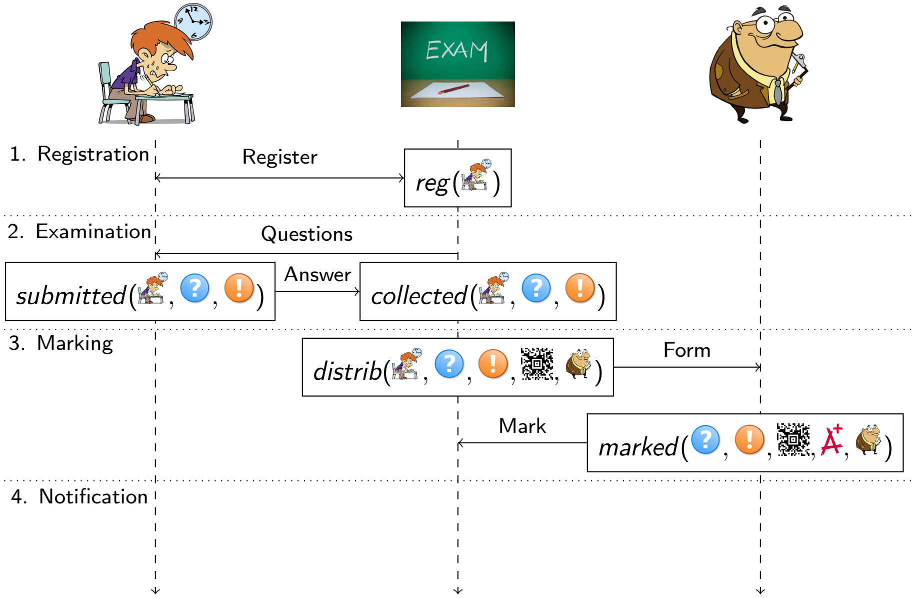

EXAM
1. Registration
Register
reg( )
2. Examination
Questions
submitted(, ,)
Answer
collected(, ,)
3. Marking
distrib(, , , ,)
Form

Model



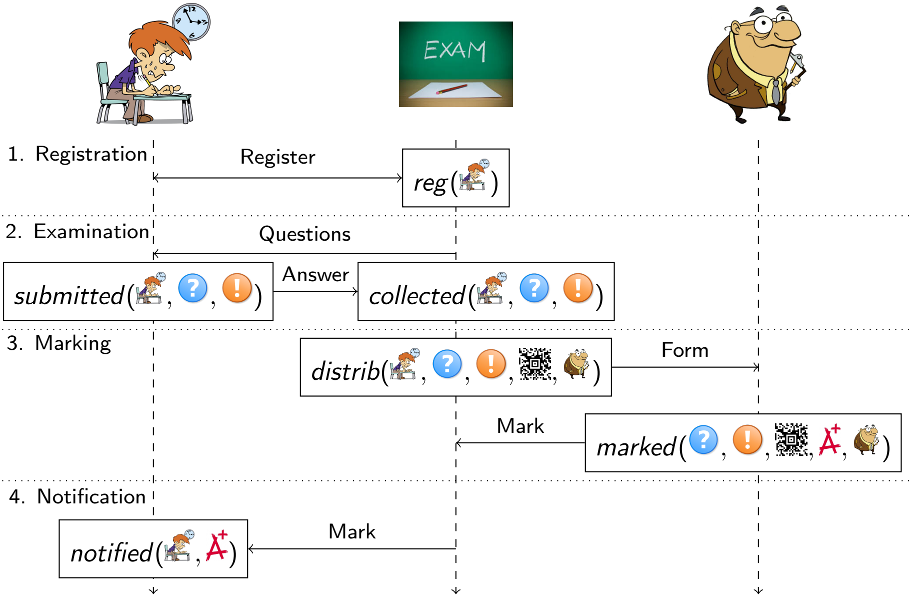
EXAM
1. Registration
Register
reg( )
2. Examination
Questions
submitted(, ,)
Answer
collected(, ,)
3. Marking
distrib(, , , ,)
Form
Mark
marked(, , , ,)

Model



EXAM
1. Registration
Register
reg( )
2. Examination
Questions
Answer
submitted(, ,)
collected(, ,)
3. Marking
distrib(, , , ,)
Form
Mark
marked(, , , ,)
4. Notification

Model


EXAM
1. Registration
Register
reg( )
2. Examination
Questions
submitted(, ,)
Answer
collected(, ,)
3. Marking
distrib(, , , ,)
Form
Mark
marked(, , , ,)
4. Notification
notified(,)
Mark

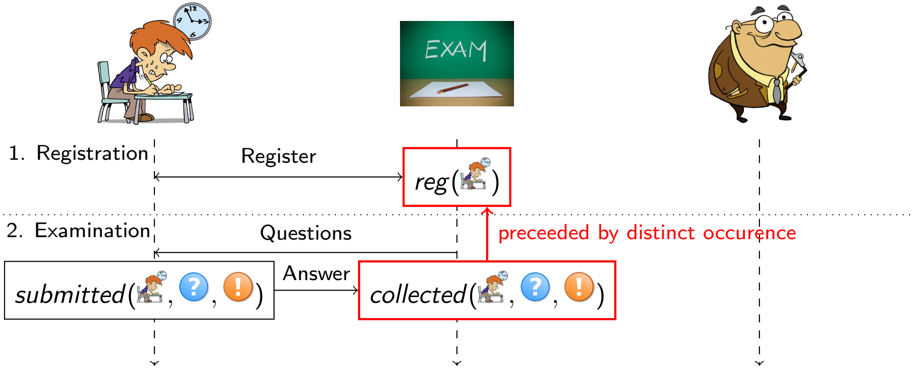
Plan

Answer Origin Authentication

All collected answers originate from registered candidates, and only one answer per candidate is accepted.

Definition:

On every trace:

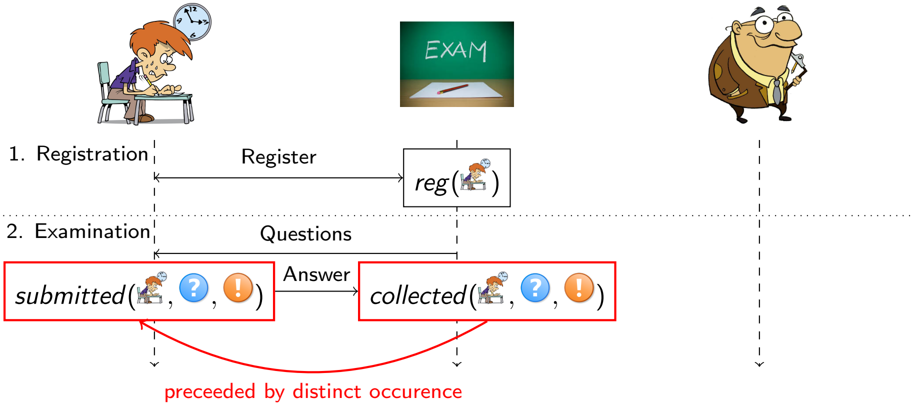


Form Authorship

Answers are collected as submitted, i.e. without modification.

Definition:

On every trace:

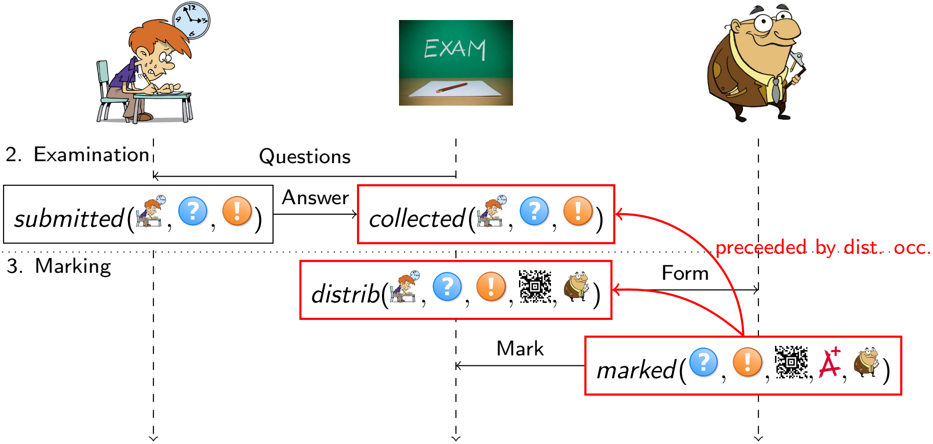


Form Authenticity

Answers are marked as collected.

Definition:

On every trace:

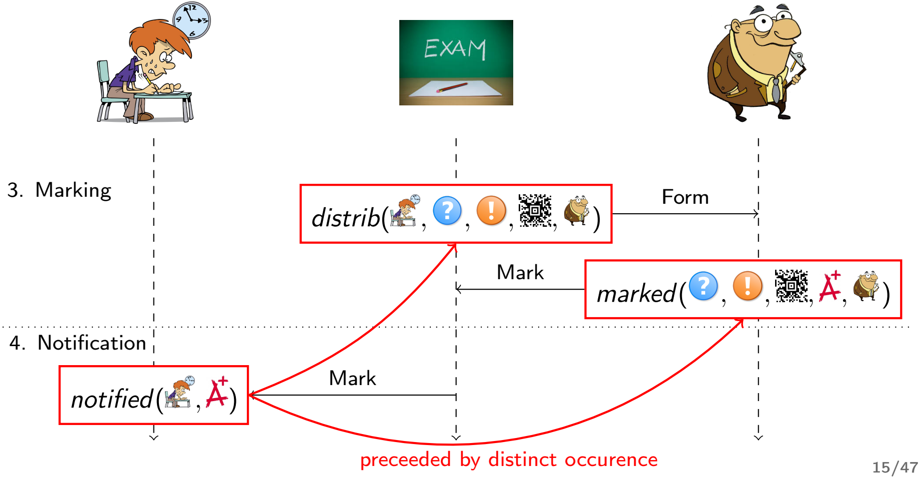


Mark Authenticity

The candidate is notified with the mark associated to his answer.

Definition:

On every trace:



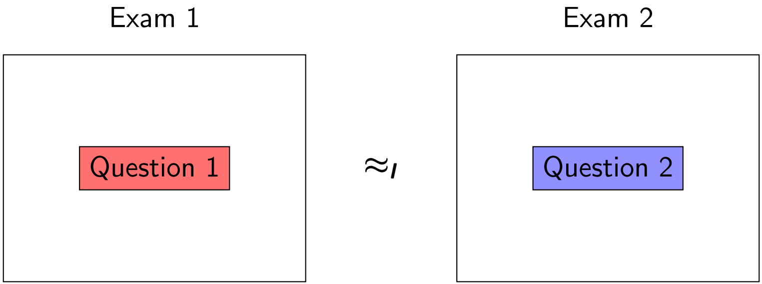
Plan

Question Indistinguishability

No premature information about the questions is leaked.

Definition:

Observational equivalence of two instances up to the end of registration phase:

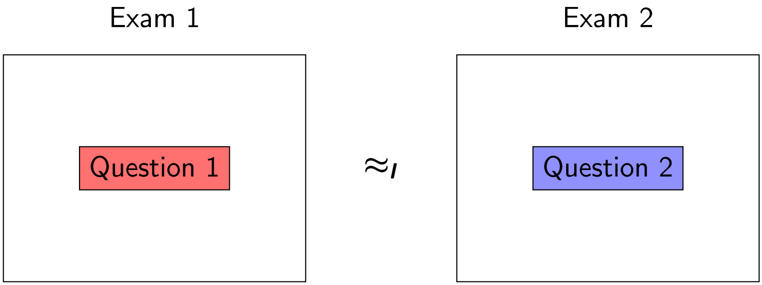


Question Indistinguishability

No premature information about the questions is leaked.

Definition:

Observational equivalence of two instances up to the end of registration phase:



Can be considered with or without dishonest candidates.

Anonymous Marking

An examiner cannot link an answer to a candidate.

Definition:

Up to the end of marking phase:

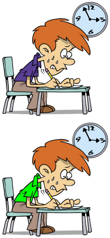
Exam 1		Exam 2
Answer 1		Answer 2
	$\approx_I$	
Answer 2		Answer 1

Anonymous Marking

An examiner cannot link an answer to a candidate.

Definition:

Up to the end of marking phase:



Exam 1		Exam 2
Answer 1		Answer 2
	$\approx_l$	
Answer 2		Answer 1

Can be considered with or without dishonest examiners and authorities.

Anonymous Examiner

A candidate cannot know which examiner graded his copy.

Definition:

	Exam 1			Exam 2	
	Answer 1	Mark 1	$\approx_l$	Answer 2	Mark 2
	Answer 2	Mark 2		Answer 1	Mark 1

Can be considered with or without dishonest candidates.

Mark Privacy

Marks are private.

Definition:

	Exam 1		Exam 2
	Answer 1 Mark 1	$\approx_I$	Answer 1 Mark 2

Can be considered with or without dishonest candidates, examiners and authorities.

Mark Anonymity

Marks can be published, but may not be linked to candidates.

Definition:

Exam 1			Exam 2	
Answer 1	Mark 1	$\approx_I$	Answer 1	Mark 2
Answer 2	Mark 2		Answer 2	Mark 1

Can be considered with or without dishonest candidates, examiners and authorities.
Implied by Mark Privacy.

Plan

Plan

Application: Huszti & Pethő's Protocol

"A Secure Electronic Exam System" [?] using

- ElGamal Encryption
- a Reusable Anonymous Return Channel (RARC) [?] for **anonymous communication**
- a network of servers providing a timed-release service using Shamir's Secret Sharing:
  A subset of servers can combine their shares to **de-anonymize a candidate** after the exam

Goal: ensure

- authentication and privacy

in presence of **dishonest**

- candidates
- examiners
- exam authorities

Results

Formal Verification with ProVerif [?]:

Property	Result	Time
Answer Origin Authentication	×	$<$ 1 s
Form Authorship	×	$<$ 1 s
Form Authenticity	×	$<$ 1 s
Mark Authenticity	×	$<$ 1 s
Question Indistinguishability	×	$<$ 1 s
Anonymous Marking	×	8 m 46 s
Anonymous Examiner	×	9 m 8 s
Mark Privacy	×	39 m 8 s
Mark Anonymity	×	1h 15 m 58 s

Main reason

Given its security definition, the **RARC**

- provides anonymity, but not necessarily secrecy
- does not necessarily provide integrity or authentication
- is only secure against **passive attackers**

Corrupted parties or active attackers can **break secrecy and anonymity**, as the following attack shows.

RARC: Mode of Operation and Attack

Input (A to RARC, destination B):

$\{ID_A, PK_A\}_{PK_{RARC}} + PoK; \{MSG\}_{PK_{RARC}}; \{ID_B, PK_B\}_{PK_{RARC}} + PoK$

RARC: Mode of Operation and Attack

Input (A to RARC, destination B):

$\{ID_A, PK_A\}_{PK_{RARC}} + PoK; \{MSG\}_{PK_{RARC}}; \{ID_B, PK_B\}_{PK_{RARC}} + PoK$

Output (RARC to B):

$$\{ID_A, PK_A\}_{PK_{RARC}} + Signature; \{MSG\}_{PK_B}$$

RARC: Mode of Operation and Attack

Input (A to RARC, destination B):

$\{ID_A, PK_A\}_{PK_{RARC}} + PoK; \{MSG\}_{PK_{RARC}}; \{ID_B, PK_B\}_{PK_{RARC}} + PoK$

Output (RARC to B):

$$\{ID_A, PK_A\}_{PK_{RARC}} + \mathit{Signature}; \{MSG\}_{PK_B}$$

Return (B to RARC, destination A):

$\{ID_B, PK_B\}_{PK_{RARC}} + PoK; \{MSG\}_{PK_{RARC}}; \{ID_A, PK_A\}_{PK_{RARC}} + \mathit{Signature}$

RARC: Mode of Operation and Attack

Input (A to RARC, destination B):

$\{ID_A, PK_A\}_{PK_{RARC}} + PoK; \{MSG\}_{PK_{RARC}}; \{ID_B, PK_B\}_{PK_{RARC}} + PoK$

Output (RARC to B):

$$\{ID_A, PK_A\}_{PK_{RARC}} + Signature; \{MSG\}_{PK_B}$$

Return (B to RARC, destination A):

$\{ID_B, PK_B\}_{PK_{RARC}} + PoK; \{MSG\}_{PK_{RARC}}; \{ID_A, PK_A\}_{PK_{RARC}} + Signature$

Attack

Input (AD to RARC, destination AD):

$\{ID_{AD}, PK_{AD}\}_{PK_{RARC}} + PoK; \{MSG\}_{PK_{RARC}}; \{ID_{AD}, PK_{AD}\}_{PK_{RARC}} + PoK$

RARC: Mode of Operation and Attack

Input (A to RARC, destination B):

$\{ID_A, PK_A\}_{PK_{RARC}} + PoK; \{MSG\}_{PK_{RARC}}; \{ID_B, PK_B\}_{PK_{RARC}} + PoK$

Output (RARC to B):

$$\{ID_A, PK_A\}_{PK_{RARC}} + Signature; \{MSG\}_{PK_B}$$

Return (B to RARC, destination A):

$\{ID_B, PK_B\}_{PK_{RARC}} + PoK; \{MSG\}_{PK_{RARC}}; \{ID_A, PK_A\}_{PK_{RARC}} + Signature$

Attack

Input (AD to RARC, destination AD):

$\{ID_{AD}, PK_{AD}\}_{PK_{RARC}} + PoK; \{MSG\}_{PK_{RARC}}; \{ID_{AD}, PK_{AD}\}_{PK_{RARC}} + PoK$

Output (RARC to AD):

$$\{ID_{AD}, PK_{AD}\}_{PK_{RARC}} + Signature; \{MSG\}_{PK_{AD}}$$

Plan

Application: Remark! Protocol

A recent protocol [?] using

- ElGamal encryption
- an **exponentiation mixnet** [?] to create **pseudonyms** based on the parties' public keys
  $\Rightarrow$ allows to encrypt and sign anonymously
- a public append-only **bulletin board**

Goal: ensure

- authentication and integrity
- privacy
- verifiability

in presence of **dishonest**

- candidates
- examiners
- exam authorities

Results

Formal Verification with ProVerif:

Property	Result	Time
Answer Origin Authentication	✓	< 1 s
Form Authorship	✓	< 1 s
Form Authenticity	✓[1]	< 1 s
Mark Authenticity	✓	< 1 s
Question Indistinguishability	✓	< 1 s
Anonymous Marking	✓	2 s
Anonymous Examiner	✓	1 s
Mark Privacy	✓	3 m 32 s
Mark Anonymity	✓	-[2]

[1] after fix

[2] implied by Mark Privacy

Plan

Conclusion

- **E-exams** are used and vulnerable to attacks
- Cryptographic protocols exist, but **lack formal verification**
- **First formal framework** for analysis of e-exams:
  - Formal model in the **applied π-calculus**
  - **Definitions** for central authentication, integrity and privacy properties
- **Automated verification in ProVerif** of two case studies:
  - Huszti & Pethő's protocol: Fails on all properties due to severe flaws in protocol design
  - Remark! protocol: Ensures all properties after one fix
- **Future work**: verifiability and accountability, analyzing implementations

Thank you for your attention!

Questions?

jannik.dreier@inf.ethz.ch

Model Definition

Definition

(E-exam protocol). *An e-exam protocol is a tuple*

$$(C, E, Q, A_1, \ldots, A_l, \tilde{n}_p),$$

where

- C *is the process executed by the candidates,*
- E *is the process executed by the examiners,*
- Q *is the process executed by the question commitee,*
- A_i*'s are the processes executed by the authorities, and*
- $\tilde{n}_p$ *is the set of private channel names.*

Model Definition cont'd

Definition

(E-exam instance). *An e-exam instance is a closed process*

$$EP = \nu\tilde{n}.(C\sigma_{id_1}\sigma_{a_1}| \dots |C\sigma_{id_j}\sigma_{a_j}|E\sigma_{id'_1}\sigma_{m_1}| \dots |E\sigma_{id'_k}\sigma_{m_k}| \\ Q\sigma_q|A_1\sigma_{dist}| \dots |A_l),$$

where

- *$\tilde{n}$ is the set of all restricted names, which includes the set of the protocol's private channels;*
- *$C\sigma_{id_i}\sigma_{a_i}$'s are the processes run by the candidates, the substitutions σ_{id_i} and σ_{a_i} specify the identity and the answers of the i^{th} candidate respectively;*
- *$E\sigma_{id'_i}\sigma_{m_i}$'s are the processes run by the examiners, the substitution $\sigma_{id'_i}$ specifies the i^{th} examiner's identity, and σ_{m_i} specifies for each possible question/answer pair the corresponding mark;*

Model Definition cont'd

Definition

(E-exam instance). *An e-exam instance is a closed process*

$$EP = \nu \tilde{n}.(C\sigma_{id_1}\sigma_{a_1}|\ldots|C\sigma_{id_j}\sigma_{a_j}|E\sigma_{id'_1}\sigma_{m_1}|\ldots|E\sigma_{id'_k}\sigma_{m_k}| \\ Q\sigma_q|A_1\sigma_{dist}|\ldots|A_l),$$

where

- *Q is the process run by the question committee, the substitution σ_q specifies the exam questions;*
- *the A_i's are the processes run by the exam authorities, the substitution σ_{dist} determines which answers will be submitted to which examiners for grading.*

Without loss of generality, we assume that A_1 is in charge of distributing the copies to the examiners.

Authentication Properties

Definition (Answer Origin Authentication)

An e-exam protocol ensures Answer Origin Authentication if, for every e-exam process EP, each occurrence of the event $\boldsymbol{collected}(\boldsymbol{id\_c}, \boldsymbol{ques}, \boldsymbol{ans})$ *is* ***preceded*** *by a distinct occurrence of the event* $\boldsymbol{reg}(\boldsymbol{id\_c})$ *on every execution trace.*

Definition (Form Authorship)

An e-exam protocol ensures Form Authorship if, for every e-exam process EP, each occurrence of the event $\boldsymbol{collected}(\boldsymbol{id\_c}, \boldsymbol{ques}, \boldsymbol{ans})$ *is* ***preceded*** *by a distinct occurrence of the event* $\boldsymbol{submitted}(\boldsymbol{id\_c}, \boldsymbol{ques}, \boldsymbol{ans})$ *on every execution trace.*

Authentication Properties cont'd

Definition (Form Authenticity)

An e-exam protocol ensures Form Authenticity if, for every e-exam process EP, each occurrence of the event $\boldsymbol{marked}(\boldsymbol{ques}, \boldsymbol{ans}, \boldsymbol{mark}, \boldsymbol{id\_form}, \boldsymbol{id\_e})$ *is* ***preceded*** *by a distinct occurrence of the events* $\boldsymbol{distrib}(\boldsymbol{id\_c}, \boldsymbol{ques}, \boldsymbol{ans}, \boldsymbol{id\_form}, \boldsymbol{id\_e})$ *and* $\boldsymbol{collected}(\boldsymbol{id\_c}, \boldsymbol{ques}, \boldsymbol{ans})$ *on every execution trace.*

Definition (Mark Authenticity)

An e-exam protocol ensures Mark Authenticity if, for every e-exam process EP, each occurrence of the event $\boldsymbol{notified}(\boldsymbol{id\_c}, \boldsymbol{mark})$ *is* ***preceded*** *by a distinct occurrence of the events* $\boldsymbol{marked}(\boldsymbol{ques}, \boldsymbol{ans}, \boldsymbol{mark}, \boldsymbol{id\_form}, \boldsymbol{id\_e})$ *and* $\boldsymbol{distrib}(\boldsymbol{id\_c}, \boldsymbol{ques}, \boldsymbol{ans}, \boldsymbol{id\_form}, \boldsymbol{id\_e})$ *on every execution trace.*

Privacy Properties

Definition (Question Indistinguishability)

An e-exam protocol ensures Question Indistinguishability if for any e-exam process EP that ends with the registration phase, any questions q_1 and q_2, we have that:
$EP_{\{id_Q\}}[Q\sigma_{q_1}]|_{reg} \approx_l EP_{\{id_Q\}}[Q\sigma_{q_2}]|_{reg}$.

Definition (Anonymous Marking)

An e-exam protocol ensures Anonymous Marking if for any e-exam process EP that ends with the marking phase, any two candidates id_1 and id_2, and any two answers a_1 and a_2, we have that:
$EP_{\{id_1,id_2\}}[C\sigma_{id_1}\sigma_{a_1}|C\sigma_{id_2}\sigma_{a_2}]|_{mark} \approx_l$
$EP_{\{id_1,id_2\}}[C\sigma_{id_1}\sigma_{a_2}|C\sigma_{id_2}\sigma_{a_1}]|_{mark}$.

Privacy Properties cont'd

Definition (Anonymous Examiner)

An e-exam protocol ensures Anonymous Examiner if for any e-exam process EP, any two candidates id_1, id_2, any two examiners id'_1, id'_2, and any two marks m_1, m_2, we have that:
$EP_{\{id_1,id_2,id'_1,id'_2,id_{A_1}\}}[C\sigma_{id_1}\sigma_{a_1}|C\sigma_{id_2}\sigma_{a_2}|E\sigma_{id'_1}\sigma_{m_1}|E\sigma_{id'_2}\sigma_{m_2}|A_1\sigma_{dist_1}] \approx_l$
$EP_{\{id_1,id_2,id'_1,id'_2,id_{A_1}\}}[C\sigma_{id_1}\sigma_{a_1}|C\sigma_{id_2}\sigma_{a_2}|E\sigma_{id'_1}\sigma_{m_2}|E\sigma_{id'_2}\sigma_{m_1}|A_1\sigma_{dist_2}]$
where σ_{dist_1} attributes the exam form of candidate id_1 to examiner id'_1 and the exam form of candidate id_2 to examiner id'_2, and σ_{dist_2} attributes the exam form of candidate id_1 to examiner id'_2 and the exam form of candidate id_2 to examiner id'_1.

Definition (Mark Privacy)

An e-exam protocol ensures Mark Privacy if for any e-exam process EP, any marks m_1, m_2, we have that:
$EP_{\{id'\}}[E\sigma_{id'}\sigma_{m_1}] \approx_l EP_{\{id'\}}[E\sigma_{id'}\sigma_{m_2}]$.

Definition (Mark Anonymity)

An e-exam protocol ensures Mark Anonymity if for any e-exam process EP, any candidates id_1, id_2, any examiner id'_1, any answers a_1, a_2 and a distribution σ_{dist} that assigns the answers of both candidates to the examiner, and two substitutions σ_{m_a} and σ_{m_b} which are identical, except that σ_{m_a} attributes the mark m_1 to the answer a_1 and m_2 to a_2, whereas σ_{m_b} attributes m_2 to the answer a_1 and m_1 to a_2, we have that:

$$EP_{\{id_1,id_2,id'_1,id_{A_1}\}}[C\sigma_{id_1}\sigma_{a_1}|C\sigma_{id_2}\sigma_{a_2}|E\sigma_{id'_1}\sigma_{m_a}|A_1\sigma_{dist}] \approx_l$$
$$EP_{\{id_1,id_2,id'_1,id_{A_1}\}}[C\sigma_{id_1}\sigma_{a_1}|C\sigma_{id_2}\sigma_{a_2}|E\sigma_{id'_1}\sigma_{m_b}|A_1\sigma_{dist}]$$

Remark! Equational Theory

$$\begin{aligned}
checkpseudo(pseudo_pub(pk(k), rce),\\
pseudo_priv(k, exp(rce))) &= true \\
decrypt(encrypt(m, pk(k), r), k) &= m \\
decrypt(encrypt(m, pseudo_pub(pk(k),\\
rce), r), pseudo_priv(k, exp(rce))) &= m \\
getmess(sign(m, k)) &= m \\
checksign(sign(m, k), pk(k)) &= m \\
checksign(sign(m, pseudo_priv(k,\\
exp(rce))), pseudo_pub(pk(k), rce)) &= m
\end{aligned}$$

Remark! Protocol

Assumption: The protocol assumes a list of eligible examiners and their public keys PK_E, and a list of eligible candidates and their public keys PK_C.

Examiner Registration

1- NET calculates $\overline{r}_e = \prod_{i=1}^{k} r_{e_i}$, $\overline{PK}_E = PK_E^{\overline{r}_e}$ and $h_e = g^{\overline{r}_e}$

2- NET publishes $sign((\overline{PK}_E, h_e), SK_{NET})$

3- E checks if $\overline{PK}_E = h_e^{SK_E}$

Candidate Registration

4- NET calculates $\overline{r}_c = \prod_{i=1}^{k} r_{c_i}$, $\overline{PK}_C = PK_C^{\overline{r}_c}$ and $h_c = g^{\overline{r}_c}$

5- NET publishes $sign((\overline{PK}_C, h_c), SK_{NET})$

6- C checks if $\overline{PK}_C = h_c^{SK_C}$

Examination

7- $EA \rightarrow C : \{sign(question, SK_{EA})\}_{\overline{PK}_C}$

8- $C \rightarrow EA : // \ C_a = \{question, answer, \overline{PK}_C\}$
$\{C_a, sign(C_a, \overline{SK_C, h_c})\}_{PK_{EA}}$

9- $EA \rightarrow C : \{C_a, sign(C_a, SK_{EA})\}_{\overline{PK_C}}$

Remark! Protocol Cont'd

Marking

10- $EA \rightarrow E : \{C_a, sign(C_a, SK_{EA})\}_{\overline{PK_E}}$

11- $E \rightarrow EA : // \ M_a = (sign(C_a, SK_{EA}), mark)$
$\{sign(M_a, \overline{SK_E, h_e})\}_{PK_{EA}}$

Notification

12- $EA \rightarrow C : \{M_a, sign(M_a, \overline{SK_E, h_e})\}_{\overline{PK}_C}$

13- $NET \rightarrow EA : \{\bar{r}_c, sign(\bar{r}_c, SK_N)\}_{PK_{EA}}$

Huszti Equational Theory

$$
\begin{aligned}
decrypt(encrypt(m, pk(k), r), k) &= m \\
getmess(sign(m, k)) &= m \\
checksign(sign(m, k), pk(k)) &= m \\
exp(exp(g, x), y) = exp(exp(g, y), x) \\
checkproof(xproof(p, p1, g, exp(g, e), e), \\
p, p1, g, exp(g, e)) &= true \\
zkpsec(zkp\_proof(exp(b, e), e), exp(b, e)) &= true
\end{aligned}
$$

Huszti's Protocol

Setup

1 - EA publishes g and $h = g^s$

2 - $Committee \rightarrow_{priv} EA$:

$\{question, \{question\}_{SSK_{committee}}, time_{x1}\}_{PK_{MIX}}$

Candidate Registration

3 - EA checks C's eligibility, and calculates $\tilde{p} = (PK_C)^s$

4 - $EA \rightarrow NET$: $\{\tilde{p}, g_C\}$

5- NET calculates $p' = \tilde{p}^\Gamma$, and $r = g_C^\Gamma$, and stores $time_{nt}$

6 - $NET \rightarrow C$: $\{p', r\}$

7 - C calculates $p = r^{SK_C}$

8 - $EA \longleftrightarrow C$: $ZKP_{eq}((p, p'), (g, h))$ //C's pseudonym: (r, p, p')

Huszti's Protocol

Examiner Registration

9 - EA checks E's eligibility, and calculates $\tilde{q} = (PK_E)^s$

10 - $EA \rightarrow E : \{\tilde{q}, g_E\}$

11 - E calculates $q' = \tilde{q}^\alpha$, $t = g_E^\alpha$, and $q = t^{SK_E}$

12 - $EA \longleftrightarrow E : ZKP_{eq}((q, q'), (g, h))$ 13 - $E \rightarrow EA : \{t, q, q', h\}$

14 - EA checks $q^s = q'$

15 - $E \longleftrightarrow EA : ZKP_{sec}(SK_E)$

16 - EA stores $\{ID_E, PK_E\}_{PK_{\boldsymbol{MIX}}}, h$

Examination

17 - $C \rightarrow EA : \{r, p, p', h\}$

18 - EA checks $p^s = p'$

19 - $C \longleftrightarrow EA : ZKP_{sec}(SK_C)$

20 - $EA \rightarrow C : \{question, \{question\}_{SSK_{\boldsymbol{committee}}}, time_{x1}\}_{PK_{\boldsymbol{MIX}}}$

21 - $C \rightarrow EA : \{r, p, \{answer\}_{PK_{\boldsymbol{MIX}}}, time_{x2}\}$

22 - $EA \rightarrow C : Hash(r, p, p', h, trans_C, question,\ time_{x1}, time_{x2}$ $\{answer\}_{PK_{\boldsymbol{MIX}}})$

Marking

23 - $EA \rightarrow E$: $\{answer\}_{PK_{MIX}}$ // Note that EA stored $\{ID_E, PK_E\}_{PK_{MIX}}, h)$

24 - $E \rightarrow EA$:
$\{mark, Hash(mark, answer), [Hash(mark, answer)]^{SK_E}, verzkp, t, q\}$

25 - $E \longleftrightarrow EA$:
$ZKP_{eq}(Hash(mark, answer), [Hash(mark, answer)]^{SK_E}), (t, q))$

Notification

26 - $EA \rightarrow NET$: $\{p'\}$ //Note that $r = g_C^{\Gamma}$, $p = PK_C^{\Gamma}$, $p' = g_C^{\Gamma s}$

27 - NET calculates $p' = \tilde{p}^{\Gamma}$

28 - $NET \rightarrow EA$: $\{p', \tilde{p}\}$

29 - EA publishes $mark, Hash(mark, answer),$
$[Hash(mark, answer)]^{SK_E}, verzkp$