SpookChain

(from AE to OAE)

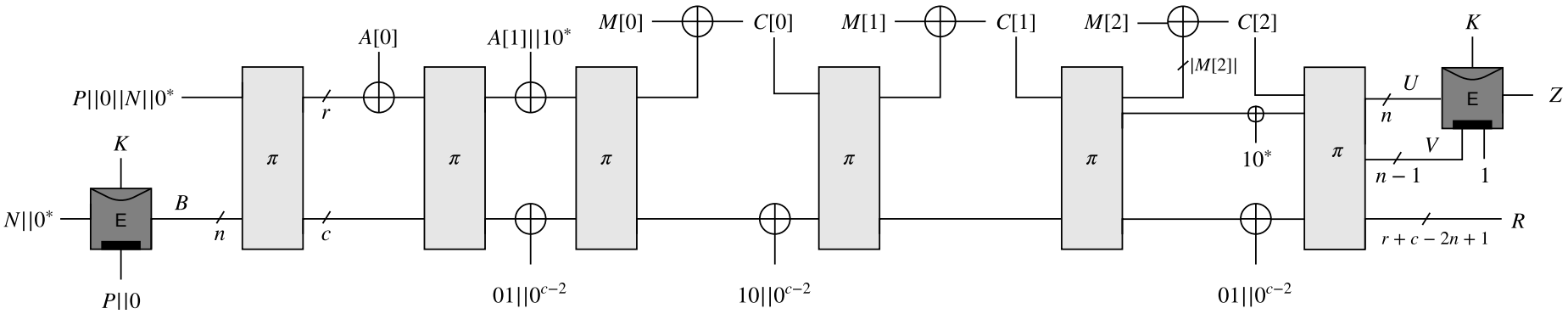
Gaëtan Cassiers, Chun Guo,
Olivier Pereira, Thomas Peters, F.-X. Standaert

UCLouvain

Louvain-la-Neuve - July 3, 2019

Online(?) Authenticated Encryption

Spook = TETSponge[Clyde,Shadow]



ciphertext $(C[0]C[1]C[2], Z) = \mathsf{Enc}_K^P(N, A[0]A[1], M[0]M[1]M[2])$

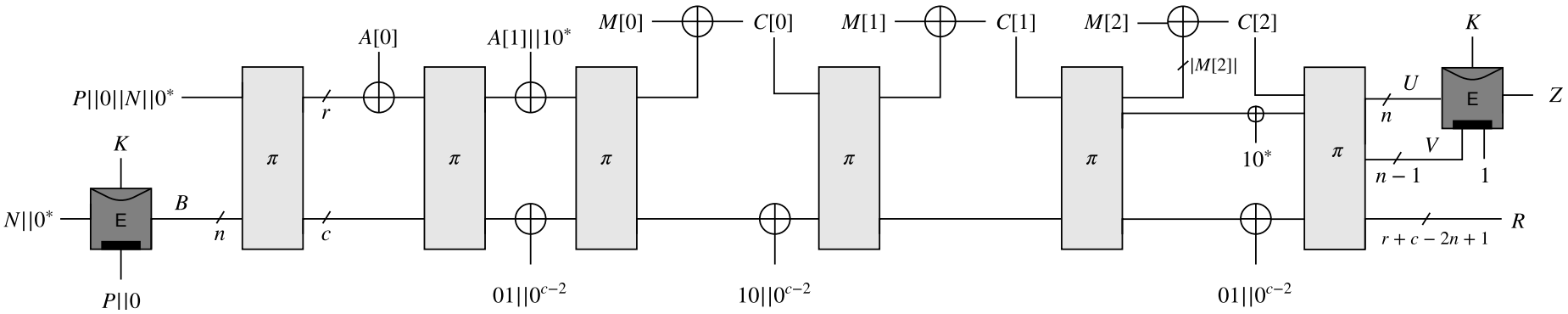
- TETSponge: encryption + decryption are already online (S1P)!
- 1-pass decryption gives 2 exclusive choices: KDF + absorb $A[0]A[1]$ +

  $\longrightarrow$ get-&-SEND $M[0]$, absorb $C[0]$ + $\cdots$ + verify Z

  $\longrightarrow$ get-&-KEEP $M[0]$, absorb $C[0]$ + $\cdots$ + verify Z + SEND $M[0]M[1]\cdots$

Online(?) Authenticated Encryption

Spook = TETSponge[Clyde,Shadow]



ciphertext $(C[0]C[1]C[2], Z) = \mathsf{Enc}_K^P(N, A[0]A[1], M[0]M[1]M[2])$

- TETSponge: encryption + decryption are already online (S1P)!
- 1-pass decryption gives 2 exclusive choices: KDF + absorb $A[0]A[1]$ +

  $\longrightarrow$ get-&-SEND $M[0]$, absorb $C[0]$ + $\cdots$ + verify Z

  $\longrightarrow$ get-&-KEEP $M[0]$, absorb $C[0]$ + $\cdots$ + verify Z + SEND $M[0]M[1]\cdots$

Online(?) Authenticated Encryption

- Spook: decryption gives 2 exclusive choices: KDF + absorb $A[0]A[1]$ +

  $\longrightarrow$ get-&-SEND $M[0]$, absorb $C[0] + \cdots +$ verify Z

  $\longrightarrow$ get-&-KEEP $M[0]$, absorb $C[0] + \cdots +$ verify $Z +$ SEND $M[0]M[1]\cdots$

- S1P simply gives a (flexible) memory/security trade-off

  $\longrightarrow$ few memory capacity + release of unverified plaintext

  $\longrightarrow$ more memory capacity + check-then-release

- **Issue**: what about very long plaintext (streaming)?

  $\longrightarrow$ Can we really choose? $\longrightarrow$ Need even better choice/flexibility

Online(?) Authenticated Encryption

- Spook: decryption gives 2 exclusive choices: KDF + absorb $A[0]A[1]$ +

  $\longrightarrow$ get-&-SEND $M[0]$, absorb $C[0] + \cdots +$ verify Z

  $\longrightarrow$ get-&-KEEP $M[0]$, absorb $C[0] + \cdots +$ verify $Z +$ SEND $M[0]M[1]\cdots$

- S1P simply gives a (flexible) memory/security trade-off

  $\longrightarrow$ few memory capacity + release of unverified plaintext

  $\longrightarrow$ more memory capacity + check-then-release

- **Issue**: what about very long plaintext (streaming)?

  $\longrightarrow$ Can we really choose? $\longrightarrow$ Need even better choice/flexibility

Online(?) Authenticated Encryption

- Spook: decryption gives 2 exclusive choices: KDF + absorb $A[0]A[1]$ +

  $\longrightarrow$ get-&-SEND $M[0]$, absorb $C[0] + \cdots +$ verify Z

  $\longrightarrow$ get-&-KEEP $M[0]$, absorb $C[0] + \cdots +$ verify Z + SEND $M[0]M[1]\cdots$

- S1P simply gives a (flexible) memory/security trade-off

  $\longrightarrow$ few memory capacity + release of unverified plaintext

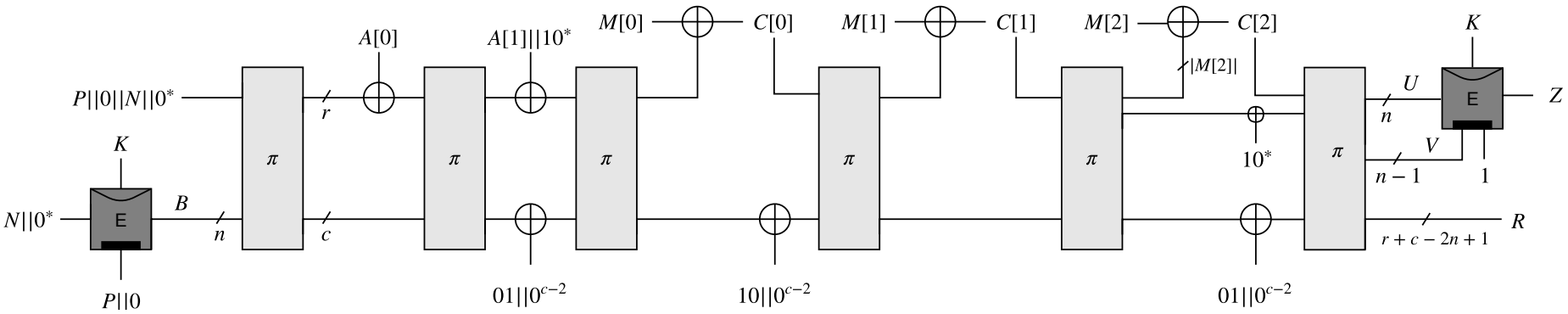
  $\longrightarrow$ more memory capacity + check-then-release

- **Issue**: what about very long plaintext (streaming)?

  $\longrightarrow$ Can we really choose? $\longrightarrow$ Need even better choice/flexibility

Online Authenticated Encryption

How to handle $M[0]M[1]M[2]M[3]M[4]\cdots M[\textit{very-long}]M[\textit{very-long}+1]\cdots$?



- OAE Goal $\longrightarrow$ security of the stream as a whole

  Somehow, chaining the AE properties of the segments

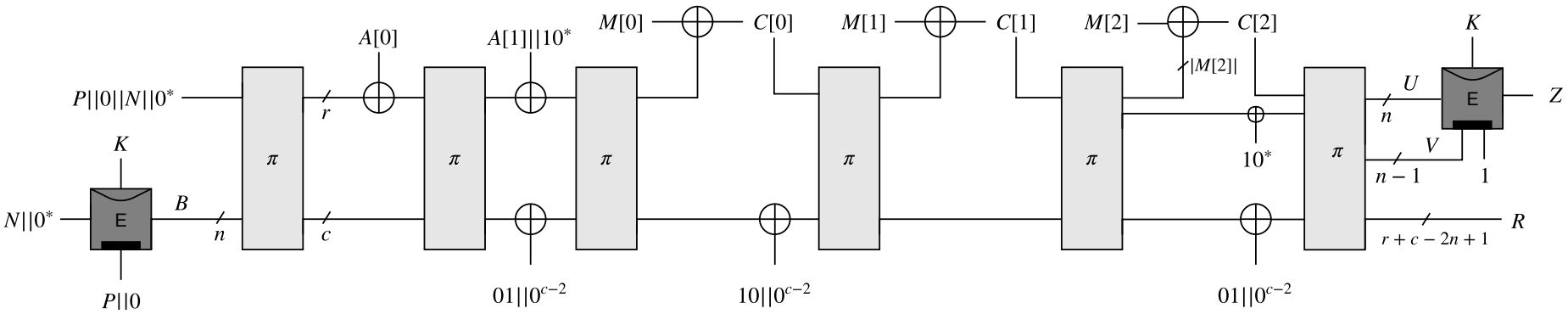
  Trade-off between space complexity and security level

- SpookChain $\longrightarrow$ *fully* online + high security

  "Partial" misuse resistance & beyond-birthday (black-box)

Online Authenticated Encryption

How to handle $M[0]M[1]M[2]M[3]M[4]\cdots M[\textit{very-long}]M[\textit{very-long}+1]\cdots$?



- OAE Goal $\longrightarrow$ security of the stream as a whole

  Somehow, chaining the AE properties of the segments

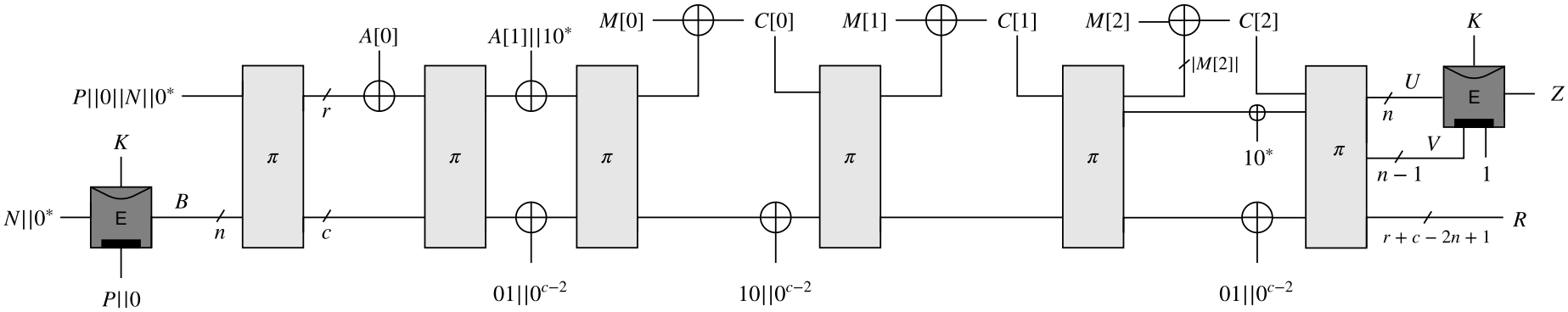
  Trade-off between space complexity and security level

- SpookChain $\longrightarrow$ *fully* online + high security

  "Partial" misuse resistance & beyond-birthday (black-box)

Online Authenticated Encryption

How to handle $M[0]M[1]M[2]M[3]M[4]\cdots M[\textit{very-long}]M[\textit{very-long}+1]\cdots$?



- Bad solution 1 $\longrightarrow$ re-use tags *("not BB-secure")*

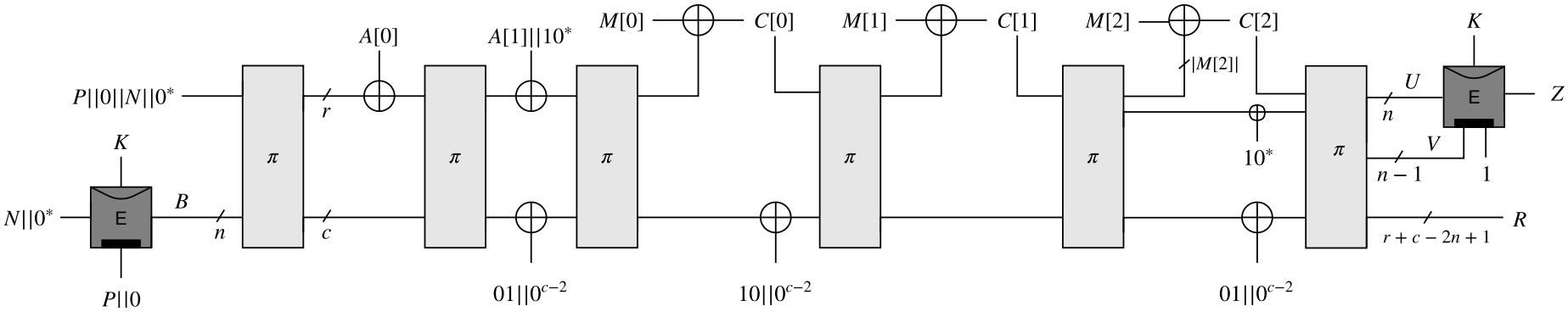
$$\mathsf{S1P}_K^P(N,1st,M[0]M[1]M[2]) + \mathsf{S1P}_K^P(Z_1,2nd,M[3]M[4]M[5]) + \cdots$$

- Bad solution 2 $\longrightarrow$ increment nonce-and-AD *(still "2 TBC-calls/segment")*

$$\mathsf{S1P}_K^P(N,1st,M[0]M[1]M[2]) + \mathsf{S1P}_K^P(N+1,2nd,M[3]M[4]M[5]) + \cdots$$

Online Authenticated Encryption

How to handle $M[0]M[1]M[2]M[3]M[4]\cdots M[\textit{very-long}]M[\textit{very-long}+1]\cdots$?



- Bad solution 1 $\longrightarrow$ re-use tags *("not BB-secure")*

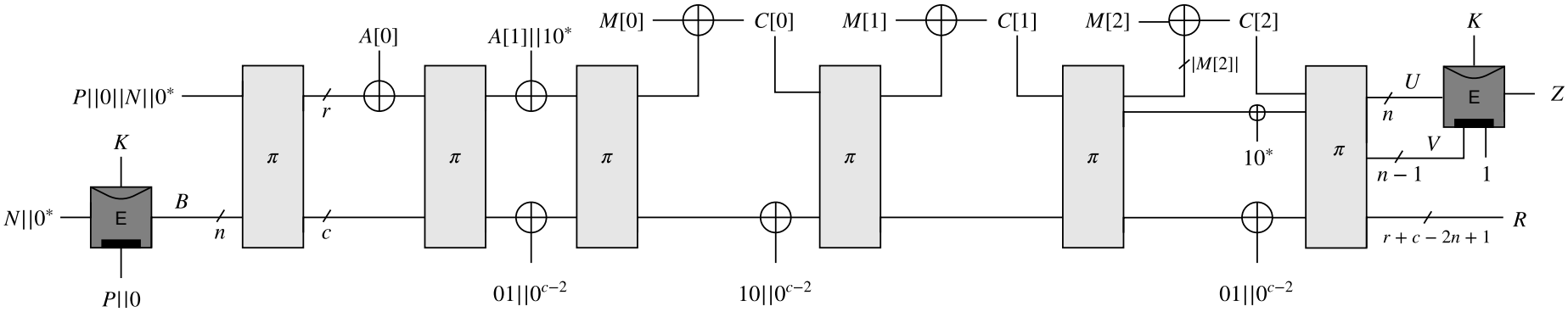
$$\mathsf{S1P}_K^P(N,1st,M[0]M[1]M[2]) + \mathsf{S1P}_K^P(Z_1,2nd,M[3]M[4]M[5]) + \cdots$$

- Bad solution 2 $\longrightarrow$ increment nonce-and-AD *(still "2 TBC-calls/segment")*

$$\mathsf{S1P}_K^P(N,1st,M[0]M[1]M[2]) + \mathsf{S1P}_K^P(N{+}1,2nd,M[3]M[4]M[5]) + \cdots$$

Online Authenticated Encryption

How to handle $M[0]M[1]M[2]M[3]M[4]\cdots M[\textit{very-long}]M[\textit{very-long}+1]\cdots$?



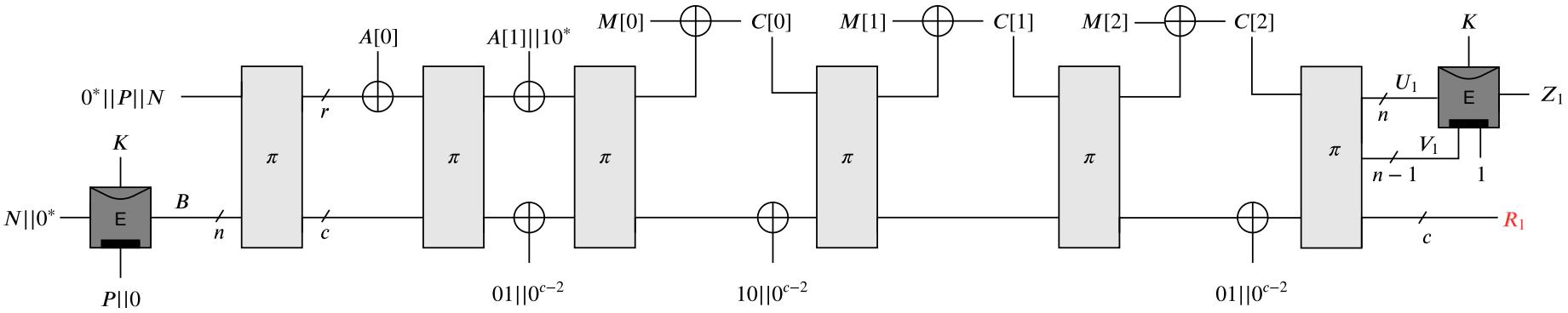
- Bad solution 1 $\longrightarrow$ re-use tags *("not BB-secure")*

$$\mathsf{S1P}_K^P(N,1st,M[0]M[1]M[2]) + \mathsf{S1P}_K^P(Z_1,2nd,M[3]M[4]M[5]) + \cdots$$

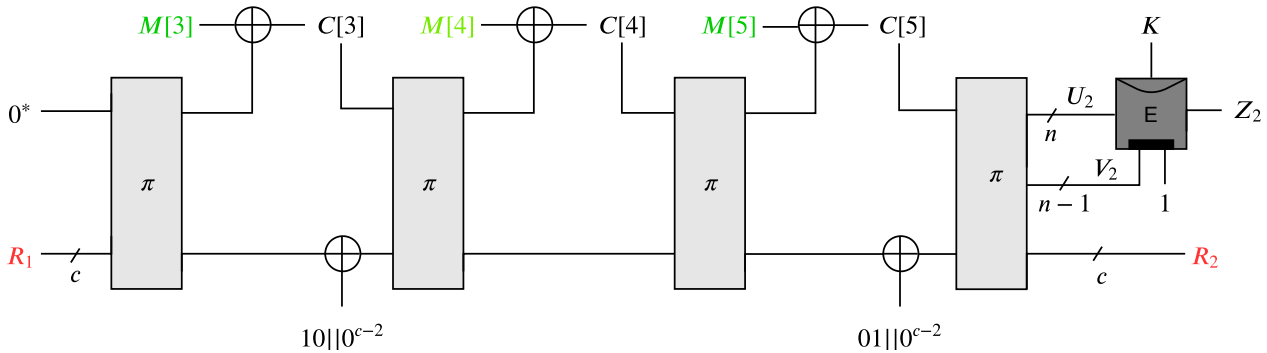
- Bad solution 2 $\longrightarrow$ increment nonce-and-AD *(still "2 TBC-calls/segment")*

$$\mathsf{S1P}_K^P(N,1st,M[0]M[1]M[2]) + \mathsf{S1P}_K^P(N{+}1,2nd,M[3]M[4]M[5]) + \cdots$$

SpookChain



Keep using the last capacity R_1 to chain the next segment $M[3]M[4]M[5]$



Advantage: High security ($c \approx 2n$) + saving TBC-call/segment

(Short) State of The Art

- Bellare-Boldyreva-Knudsen-Namprempre (Crypto'01): "first principle"
  Constant memory, block-size oriented, "online cipher"
- Fleischmann-Forler-Lucks (FSE'12): McOE
  Family of online AE, block-size oriented, security OAE1
- Hoang-Reyhanitabar-Rogaway-Vizar (Crypto'15): generalization
  New syntax, OAE2 ($\approx$ best possible), 1-pass: nAOE, dOAE
- Bertoni-Daemen-Peeters-VanAssche (SAC'11): Intermediate tag
  Duplexing the sponge, single-pass, premises of dOAE, less formalism

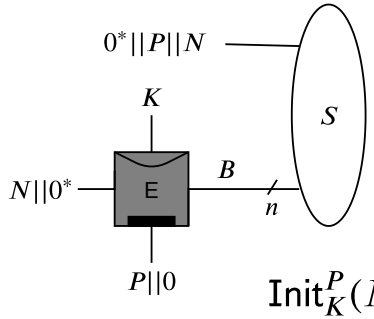
(Short) State of The Art

- Bellare-Boldyreva-Knudsen-Namprempre (Crypto'01): "first principle"
  Constant memory, block-size oriented, "online cipher"
- Fleischmann-Forler-Lucks (FSE'12): McOE
  Family of online AE, block-size oriented, security OAE1
- Hoang-Reyhanitabar-Rogaway-Vizar (Crypto'15): generalization
  New syntax, OAE2 ($\approx$ best possible), 1-pass: nAOE, dOAE
- Bertoni-Daemen-Peeters-VanAssche (SAC'11): Intermediate tag
  Duplexing the sponge, single-pass, premises of dOAE, less formalism

(Short) State of The Art

- Bellare-Boldyreva-Knudsen-Namprempre (Crypto'01): "first principle"
  Constant memory, block-size oriented, "online cipher"
- Fleischmann-Forler-Lucks (FSE'12): McOE
  Family of online AE, block-size oriented, security OAE1
- Hoang-Reyhanitabar-Rogaway-Vizar (Crypto'15): generalization
  New syntax, OAE2 ($\approx$ best possible), 1-pass: nAOE, dOAE
- Bertoni-Daemen-Peeters-VanAssche (SAC'11): Intermediate tag
  Duplexing the sponge, single-pass, premises of dOAE, less formalism

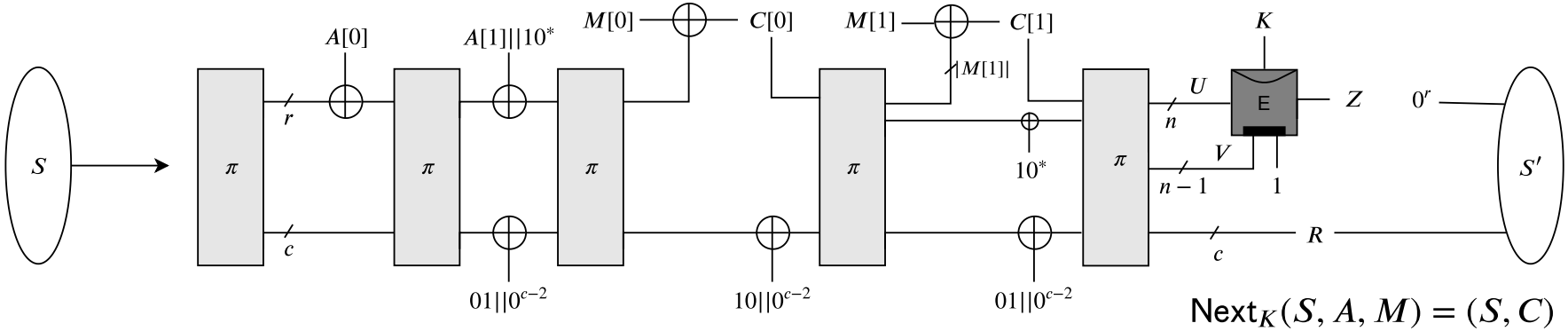
SpookChain



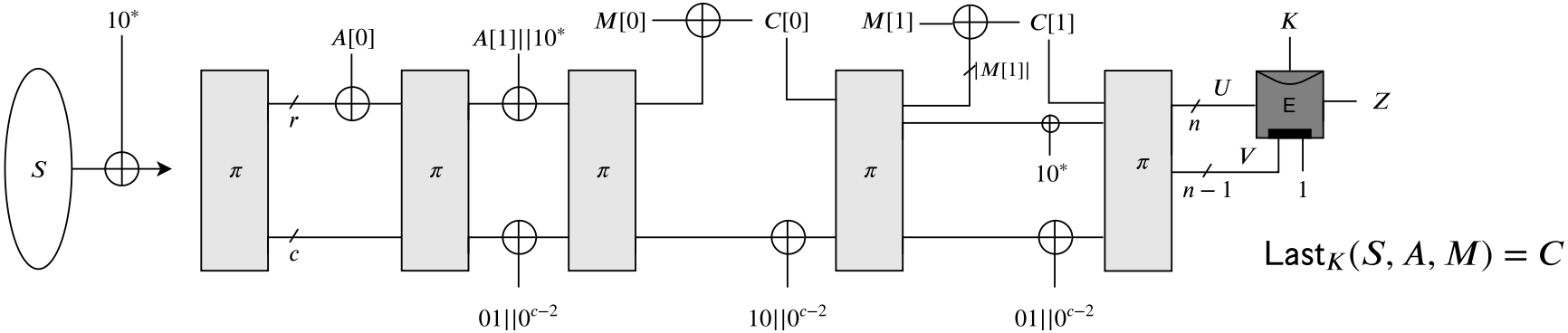
$\text{Init}_K^P(N) = S$

Chaining segmented-AE

Init Next Last



$\text{Next}_K(S, A, M) = (S, C)$



$\text{Last}_K(S, A, M) = C$

Conclusion

New OAE-like Definition

- dOAE in the multi-key setting
- Achievable for **fully** online design
- Room for improvement: black-box & leakage (nOAE, OAE-CIML)

New Construction

- From TETSponge (Spook) to TETSpongeChain (SpookChain)
- **Security**: Beyond-Birthday secure + **Efficiency**: 1-TBC per segment
- Same structure than Spook (extension, no change)

Conclusion

New OAE-like Definition

- dOAE in the multi-key setting
- Achievable for **fully** online design
- Room for improvement: black-box & leakage (nOAE, OAE-CIML)

New Construction

- From TETSponge (Spook) to TETSpongeChain (SpookChain)
- **Security**: Beyond-Birthday secure + **Efficiency**: 1-TBC per segment
- Same structure than Spook (extension, no change)

Thank you!

Questions?