Proactive Secret Sharing with a Dishonest Majority

Shlomi Dolev*, Karim ElDefrawy**, Joshua Lampkins**,
Rafail Ostrovsky***, Moti Yung****

* Ben-Gurion University

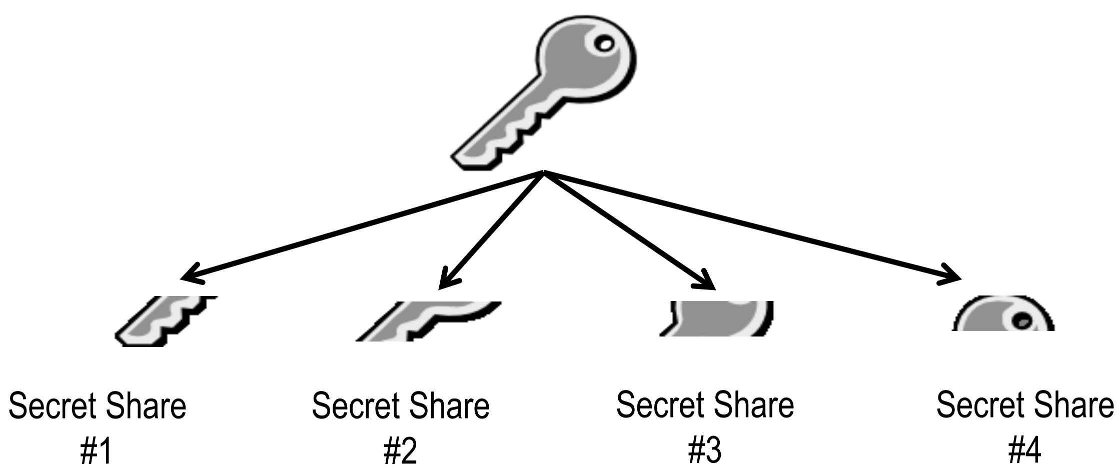
** Hughes Research Labs (HRL)

*** University of California Los Angeles (UCLA)

**** Snapchat and Columbia University

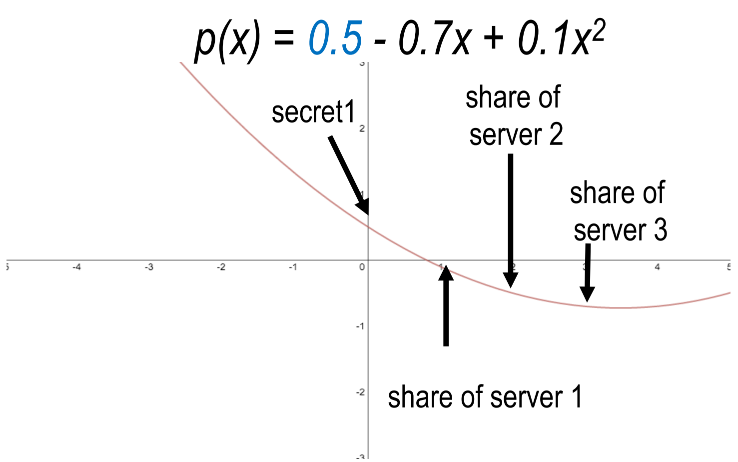
Secret Sharing (1/2)

- A ***t* out of *n*** secret sharing scheme **shares** a secret among *n* parties.
- Any *t* + 1 parties can combine their shares to **reconstruct** the secret.
- With only ***t* of the *n* shares one does not learn** any information about the secret.
- Invented independently by Blakely and Shamir (1979).



Secret Sharing (2/2)

- **Shamir's Technique**: store secret in constant term of degree t polynomial to tolerate up to t leaked shares (called $t + 1$ out of n)

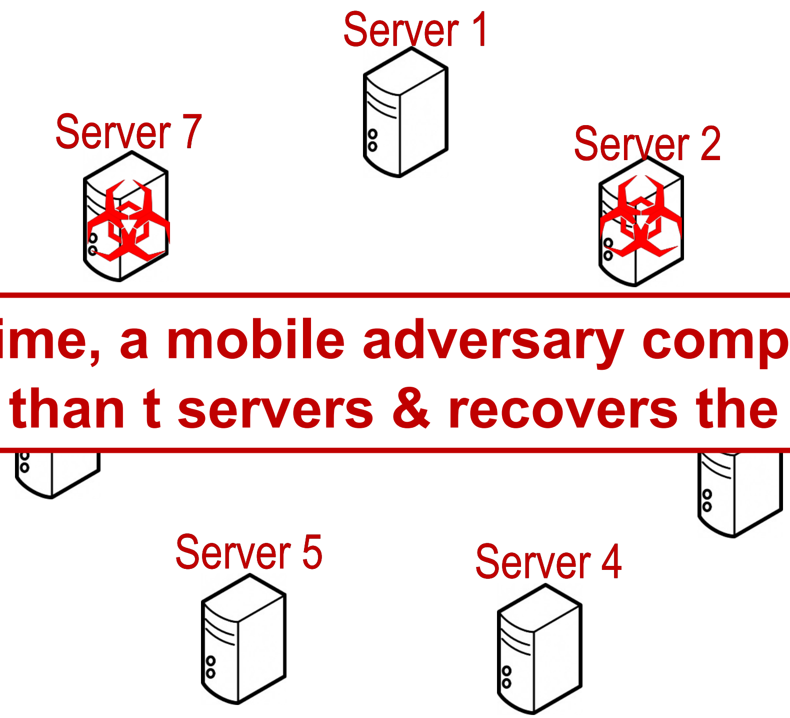


- **Secret Sharing Involves Two Algorithms:**

  i. ***Share:*** for secret s, pick random coefficients $a_1 \dots a_t$ & set $a_0 = s$ and $p(x) = a_0 + a_1 x + a_2 x^2 + \dots a_t x^t$ distribute shares as $p(1), p(2) \dots f(n)$ to the n parties

  ii. ***Open/Reconstruct:*** from $p(1), p(2) \dots p(t+1)$ interpolate $p(x)$ and recover secret as $p(0) = a_0 = s$

Mobile Adversaries



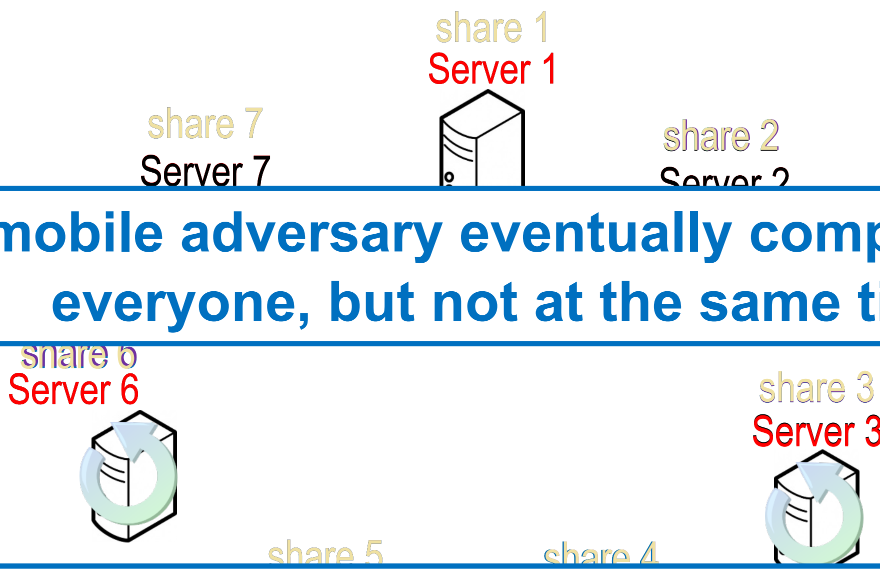
Over time, a mobile adversary compromises more than t servers & recovers the secret!

Shares Collected by Adversary

Share of Server 2
Share of Server 7
Share of Server 4
Share of Server 5
Share of Server 3
Share of Server 1
Share of Server 6

Proactive Security

Shares with different colors are from different time epochs and can **NOT** be combined.



A mobile adversary eventually compromises everyone, but not at the same time!

Shares Collected by Adversary

share 2 – Epoch 1
share 7 – Epoch 1
share 4 – Epoch 2
share 5 – Epoch 2
share 3 – Epoch 3
share 1 – Epoch 3
share 6 – Epoch 4

Proactively refresh/rerandomize shares on servers, and randomly reboot servers to a pristine state and recover their shares.

Relevance of Proactive Security Model

- **Proactively secure protocols for various cryptographic primitives were developed since 90s:**
  - **Proactive secure multi-party computation [OY91, BELO14, BELO15].**
  - **Proactive encryption/signature schemes [FGMY97a, FGMY97b, Rab98, CGJ+99, FMY01, Bol03, JS05, JO08, ADN06].**
  - **Proactive secret sharing [WWW02, ZSvR05, CKLS02, Sch07, HJKY95, DELOY16].**

Mixed Adversaries Model

- **Threshold of corruptions is defined by (A^*, P^*):**
  - ***Set of Passive Corruptions (P^*):*** semi-honest, follows protocols but tries to violate privacy
  - ***Set of Active Corruptions (A^*):*** *fully malicious, can deviate arbitrarily from protocols*
- **Each active corruption is also a passive corruption ($A^* \sqsubseteq P^*$)**
- **Multi-threshold:**
  - ***Correctness (T^c):*** threshold for which correctness is ensured
  - ***Secrecy (T^s):*** threshold for which secrecy is ensured
  - ***Robustness (T^r):*** threshold for which robustness is ensured

Our Result

Paper	Network Model	Dynamic Groups	Security	Threshold	Communication (amortized)
[WWW02]	Synch.	No	Crypto.	$t/n < 1/2$	exp(n)
[ZSvR05]	Asynch.	No	Crypto	$t/n < 1/3$	exp(n)
[CKLS02]	Asynch.	No	Crypto	$t/n < 1/3$	$O(n^4)$
[Sch07]	Asynch.	Yes	Crypto	$t/n < 1/3$	$O(n^4)$
[OY91]	Synch.	No	Statistical	$t/n < 1/3$	$O(n^3)$
[HJKY95]	Synch.	No	Crypto	$t/n < 1/2$	$O(n^2)$
[BELO14]	Synch.	No	Perfect / Statistical	$t/n < 1/3-\varepsilon$ / $t/n < 1/2-\varepsilon$	O(1)
[BELO15]	Synch.	Yes	Perfect / Statistical	$t/n < 1/3-\varepsilon$ / $t/n < 1/2-\varepsilon$	O(1)
[DELOY16]	Synch.	No	Crypto (homomorphic commitments)	$t < n - r$ (passive only) $t < n/2 - r$ (active) $t < n - k - r$ (mixed adversaries) t = total corruptions k = active corruptions r = number of nodes reset in parallel	$O(n^4)$

[DELOY16] Proactive Secret Sharing (PSS) where t could be > n/2, when k = 0 (i.e., passive corruptions only) $t < n - r$, r = 1 if nodes will be reset serially.

Background: Gradual Secret Sharing

- **First introduced in [HML13] for mixed adversaries (a mix of passive and active corruptions)**

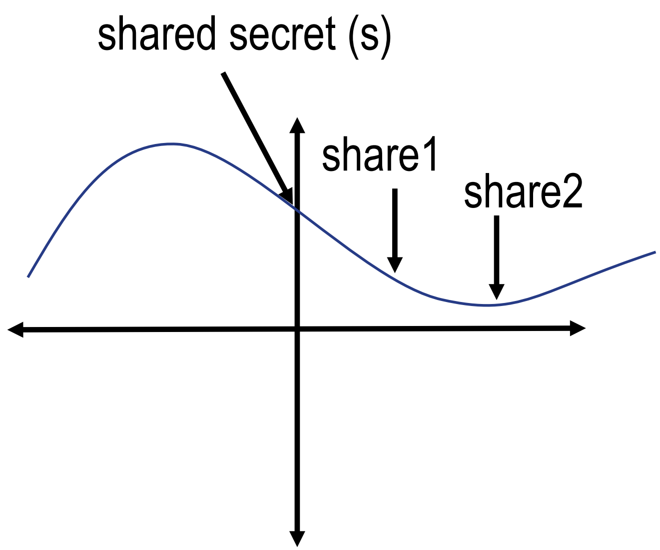
- **Secure against a dishonest majority with identifiable aborts**

- **Share:** A d-gradual secret sharing of a secret s does the following:
  - Split s into d random summands, $s = \sum_{i=1}^{d} s_i$
  - Share each s_i with a random polynomial of degree *i*

- **Reconstruct:** to recover s shared with a d-gradual secret sharing:
  - Reconstruct the *d* polynomials in decreasing order (from *d* down to *1*)
  - For polynomial *i* if less than *i+1* parties are honest abort and identify misbehaving parties

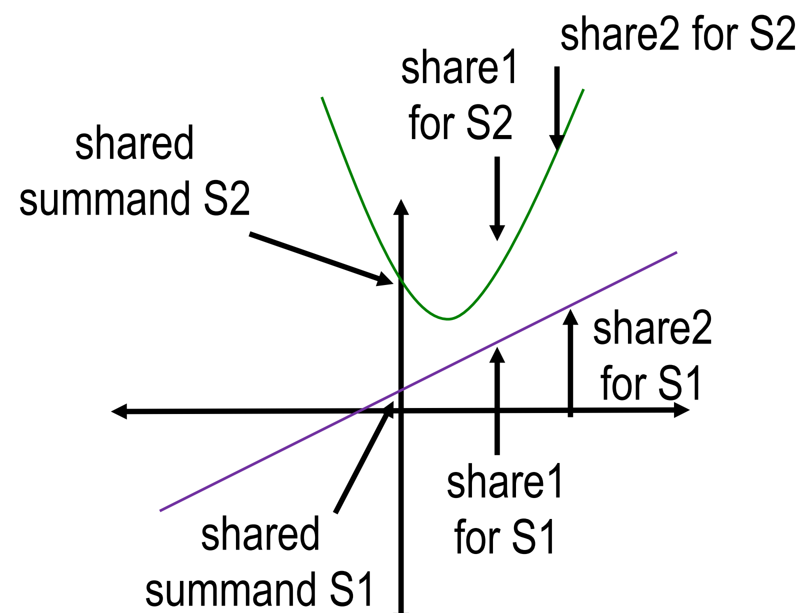
Single vs. Gradual Secret Sharing

Linear Sharing [Sha79]



- Secret is stored as a free term in a polynomial of degree t
- Confidentiality lost if t+1 parties compromised, typically t < n/2
- Robust

Gradual Sharing [HML13]



- Confidentiality is not lost as long as at most d < n parties are compromised
- Non-robust with active adversaries

PSS Blueprint for Dishonest Majority

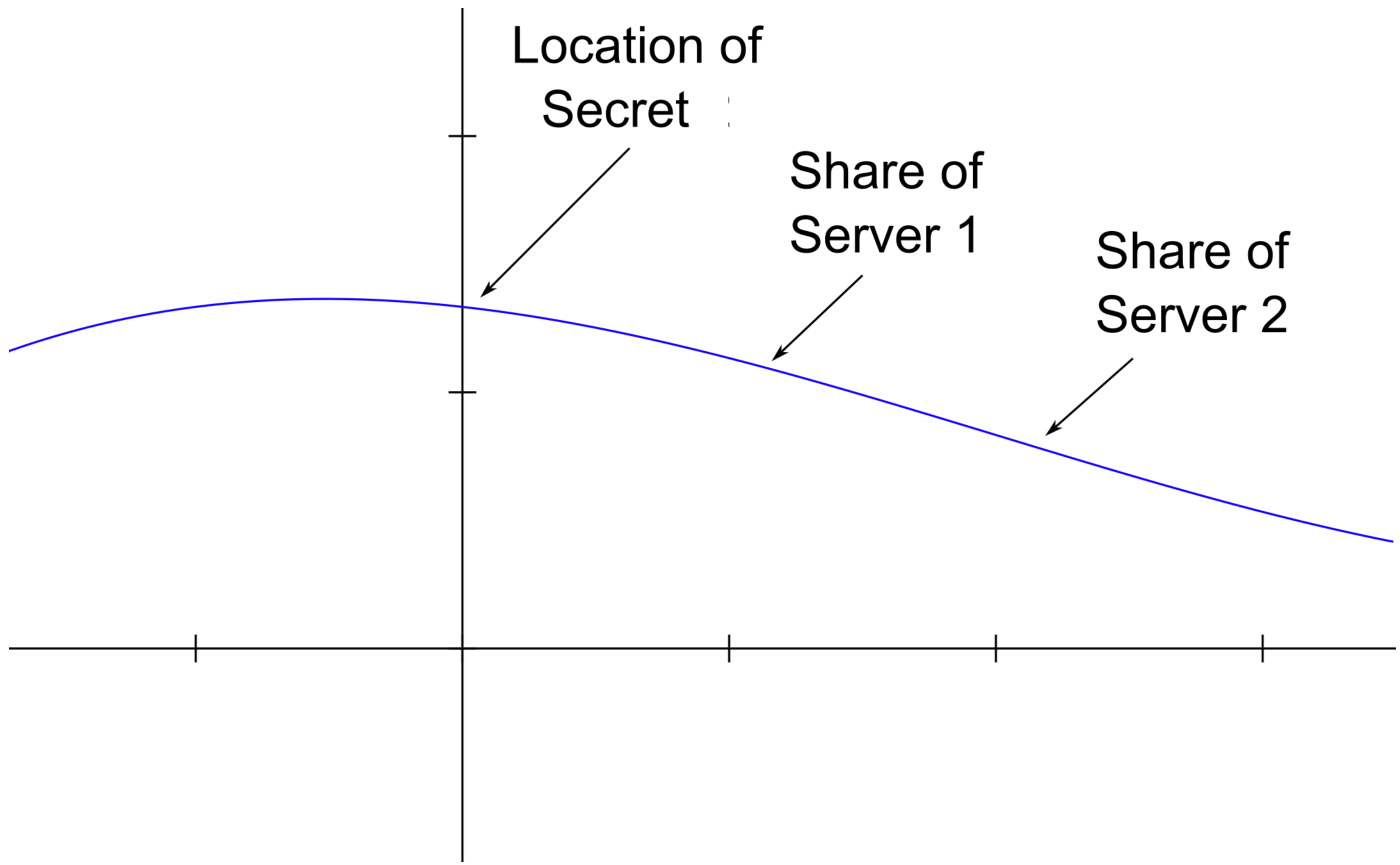
- **Use Gradual Secret Sharing with a maximum degree less than $d = n - r$ where r is the number of parties that can be rebooted in parallel.**

- **Proactivizing Gradual Secret Sharing by developing two protocols with same security guarantees against mixed adversaries and dishonest majority:**

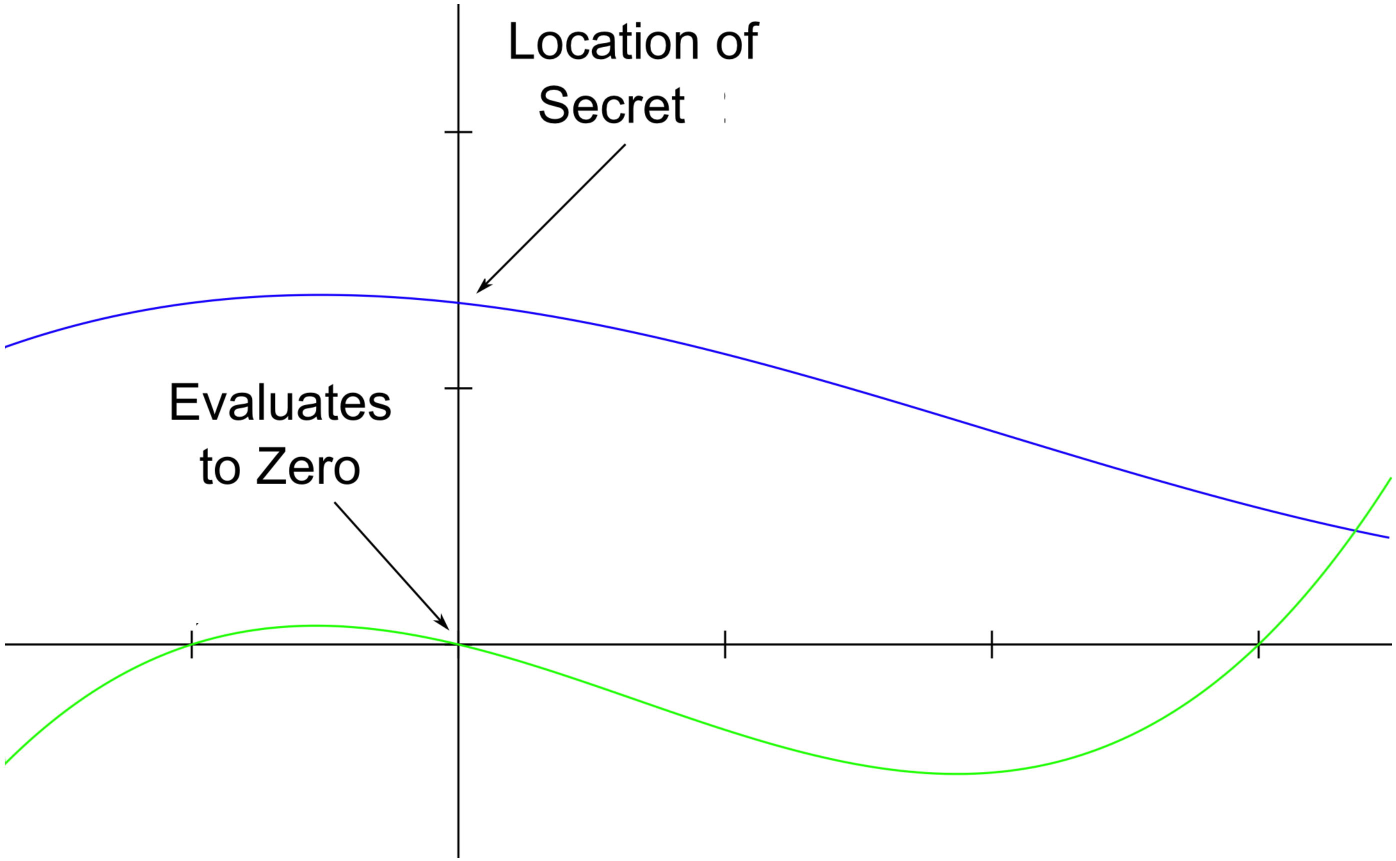
1. **Refresh:** distributed rerandomization of shares

2. **Recovery:** distributed recovery of shares (for rebooted nodes)

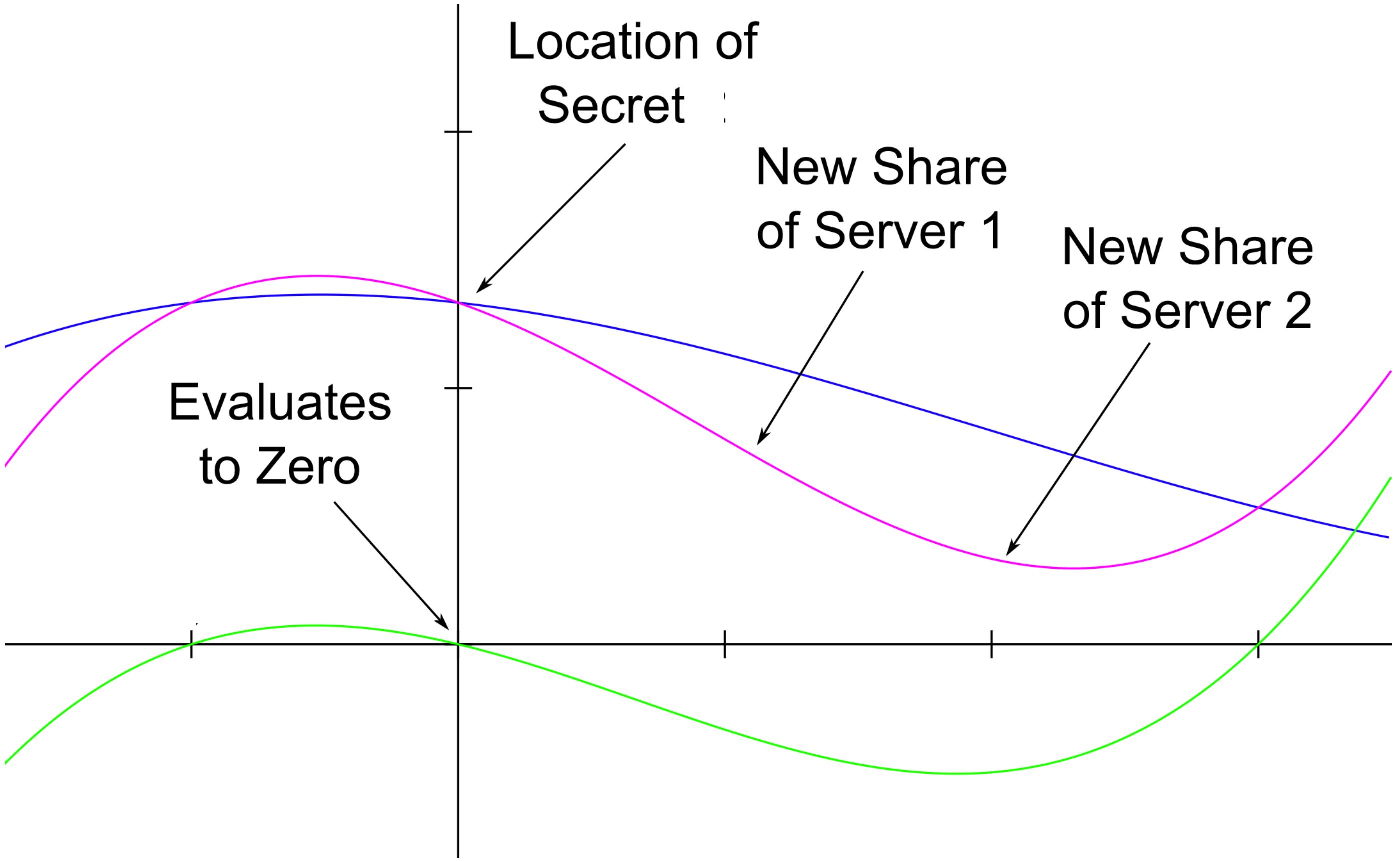
Refreshing Shares of a Summand (1/3)



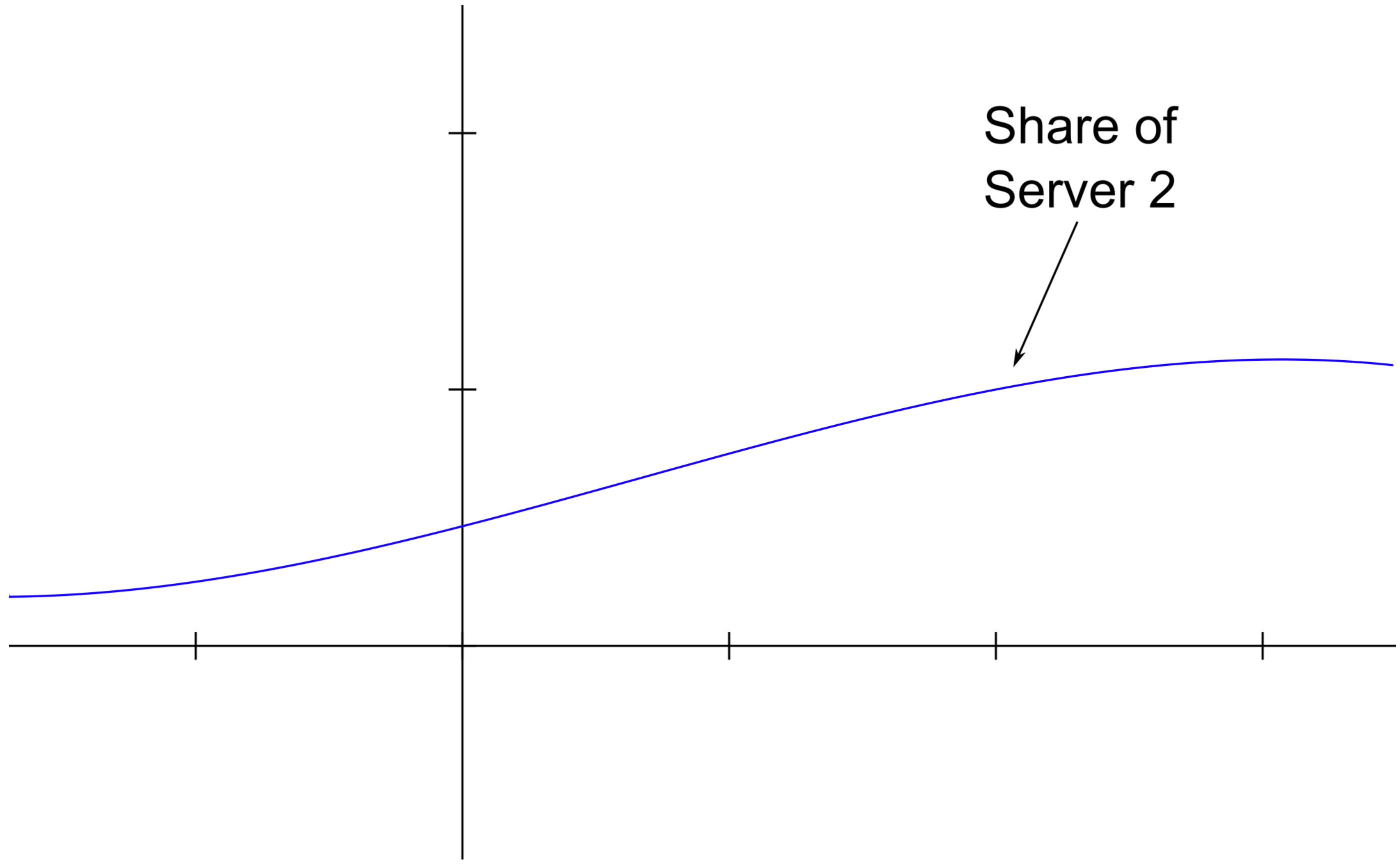
Refreshing Shares of a Summand (2/3)



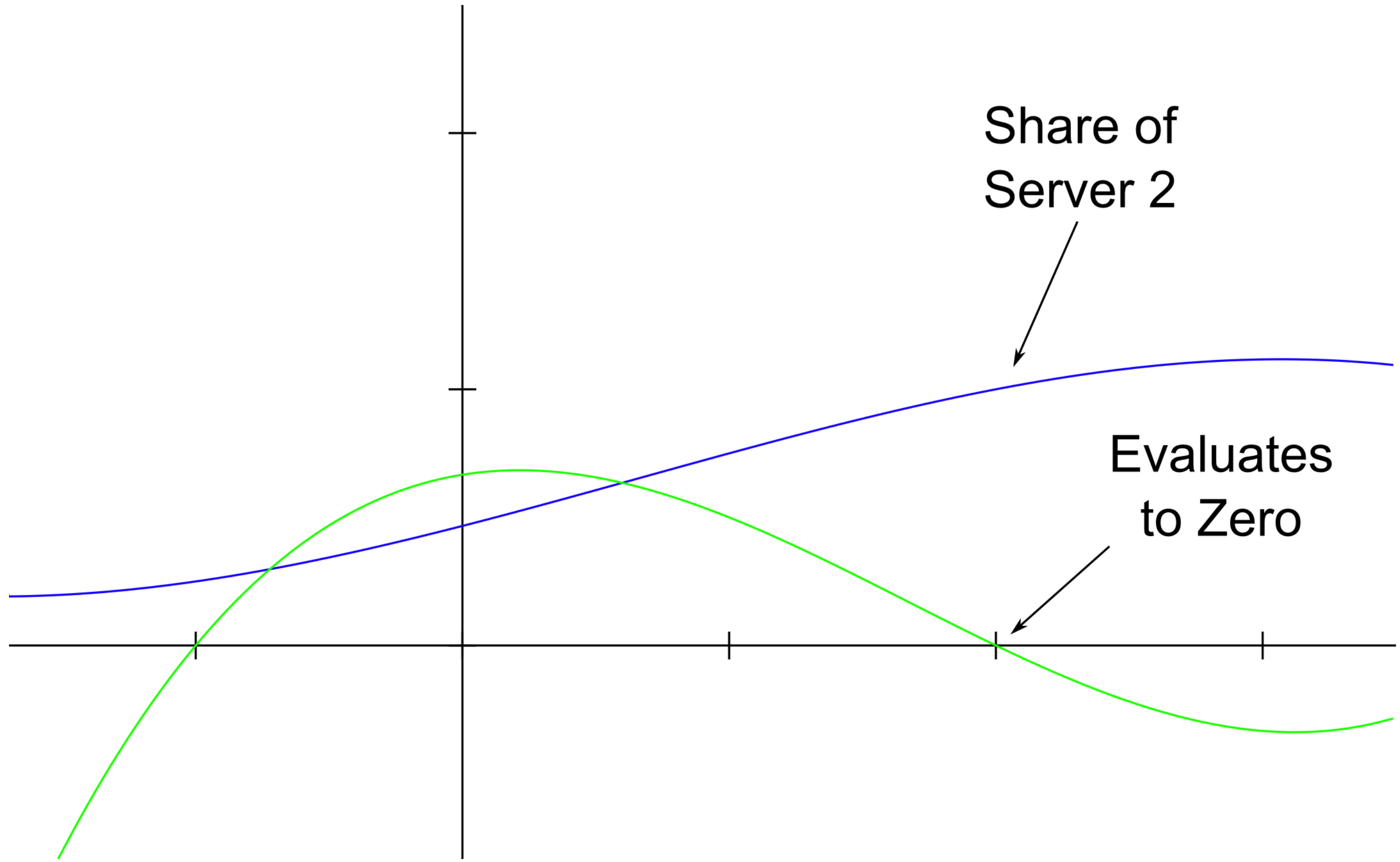
Refreshing Shares of a Summand (3/3)



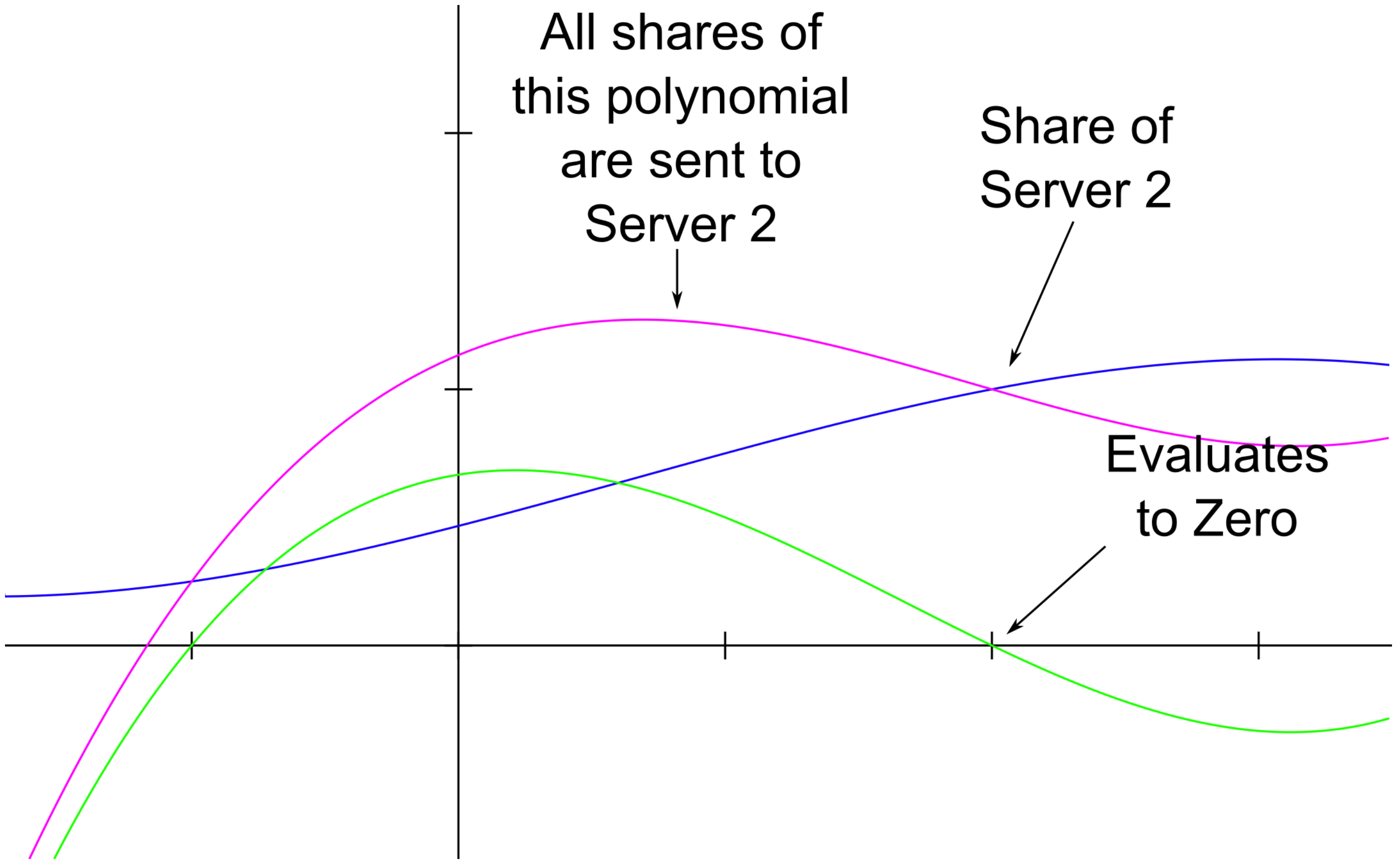
Recovering Shares of a Summand (1/3)



Recovering Shares of a Summand (2/3)



Recovering Shares of a Summand (3/3)



Main Theorem

- **For r = 1 (rebooting nodes in series) we get the highest thresholds.**

Theorem:

- Given a gradual secret sharing parameter $d < n - \mathrm{k} - 1$ there exists a computationally secure (T^s,T^r,T^c)-secure PSS scheme, utilizing a computationally secure homomorphic commitment scheme, for mixed adversaries characterized by (A^*,P^*) where $A^* \subseteq P^*$.

- The PSS scheme ensures secrecy if $|P^*| \leq d$, is robust against $|A^*| \leq k$ if $d < n-k-1$ and $|P^*| \leq d$, and is correct with agreement on aborts if $|P^*| \leq d \wedge |P^*|+|A^*| \leq n-2$.

Proof Sketches

- **Since this is only a SS, prove correctness and security as properties of the SS scheme**
- **Can be formalized to provide full simulator showing that view in real world ~ view ideal world**

- ***Secrecy:*** straightforward because of degree of polynomial

- ***Robustness:*** given a polynomial with degree less than n – r, have r redundant points so can reconstruct without them

- ***Correctness (with agreement on aborts):*** prove by contradiction by breaking correctness of PSS scheme to security of underlying commitment scheme

Future Work

- **Efficient Communication:** can communication be reduced to O(n) or even O(1)?

- **Other Blueprints:** Using a single polynomial with degree n – r – 1 and ZK proofs (constant size) can probably shave a factor *n* from communication.

- **Dynamic Groups:** extend the new PSS to dynamic groups with dishonest majority.

- ***(In Progress)*** **Extend to Proactive Secure Multiparty Computation:** perform computation with proactive refresh with similar thresholds, i.e., with a dishonest majority. Currently all proactive MPC protocols are for honest majority ($t < n/2$)**.**

Questions?