Securing Cloud Data under Key Exposure

ABSTRACT:

Recent news reveal a powerful attacker which breaks data confidentiality by acquiring cryptographic keys, bymeans of coercion or backdoors in cryptographic software. Once the encryption key is exposed, the only viable measure topreserve data confidentiality is to limit the attacker's access to the ciphertext. This may be achieved, for example, by spreadingciphertext blocks across servers in multiple administrative domains—thus assuming that the adversary cannot compromise allof them. Nevertheless, if data is encrypted with existing schemes, an adversary equipped with the encryption key, can stillcompromise a single server and decrypt the ciphertext blocks stored therein. In this paper, we study data confidentialityagainst an adversary which knows the encryption key and has access to a large fraction of the ciphertext blocks. To this end,we propose Bastion, a novel and efficient scheme that guarantees data confidentiality even if the encryption key is leakedand the adversary has access to almost all ciphertext blocks. We analyze the security of Bastion, and we evaluate itsperformance by means of a prototype implementation. We also discuss practical insights with respect to the integration ofBastion in commercial dispersed storage systems. Our evaluation results suggest that Bastion is well-suited for integration inexisting systems since it incurs less than 5% overhead compared to existing semantically secure encryption modes.

EXISTING SYSTEM:

- If the encryption key is exposed, the only viablemeans to guarantee confidentiality is to limit the adversary'saccess to the ciphertext, e.g., by

spreading itacross multiple administrative domains, in the hopethat the adversary cannot compromise all of them.

- However, even if the data is encrypted and dispersedacross different administrative domains, an adversaryequipped with the appropriate keying material cancompromise a server in one domain and decrypt ciphertextblocks stored therein.
- Ramp schemes constitute a trade-off between thesecurity guarantees of secret sharing and the efficiencyof information dispersal algorithms. A ramp schemeachieves higher "code rates" than secret sharing and features two thresholds t1, t2. At least t2 shares arerequired to reconstruct the secret and less than t1 sharesprovide no information about the secret; a number ofshares between t1 and t2 leak "some" information.
- Resch et al. combine AONT andinformation dispersal to provide both fault-toleranceand data secrecy, in the context of distributed storagesystems. In existing system, however, an adversary which knowsthe encryption key can decrypt data stored on singleservers.

DISADVANTAGES OF EXISTING SYSTEM:

- ExistingAON encryption schemes, however, require *at least* tworounds of block cipher encryptions on the data: one preprocessinground to create the AONT, followed by anotherround for the actual encryption. Notice that theserounds are sequential, and cannot be parallelized.
- Thisresults in considerable—often unacceptable—overheadto encrypt and decrypt large files.

- On the other hand,Bastion requires only one round of encryption—whichmakes it well-suited to be integrated in existing dispersedstorage systems.

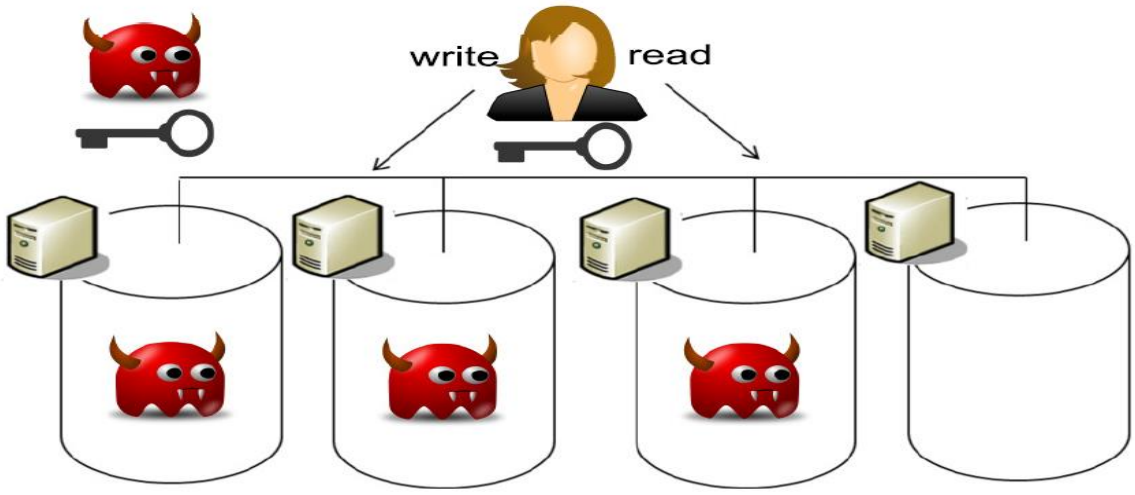
PROPOSED SYSTEM:

- In this paper, we study data confidentiality againstan adversary which knows the encryption key and hasaccess to a large fraction of the ciphertext blocks. Theadversary can acquire the key either by exploiting flawsor backdoors in the key-generation software, orby compromising the devices that store the keys (e.g.,at the user-side or in the cloud). To counter such an adversary, we propose Bastion, anovel and efficient scheme which ensures that plaintextdata cannot be recovered as long as the adversaryhas access to at most all but *two* ciphertext blocks,even when the encryption key is exposed.
- Bastionachieves this by combining the use of standard encryptionfunctions with an efficient linear transform.In this sense, Bastion shares similarities with the notionof all-or-nothing transform. An AONT is not anencryption by itself, but can be used as a pre-processingstep before encrypting the data with a block cipher.This encryption paradigm—called AON encryption—was mainly intended to slow down brute-force attackson the encryption key. However, AON encryption canalso preserve data confidentiality in case the encryptionkey is exposed, as long as the adversary has accessto at most all but one ciphertext blocks.

ADVANTAGES OF PROPOSED SYSTEM:

- We evaluate the performance of Bastion in comparisonwith a number of existing encryption schemes. Ourresults show that Bastion only incurs a negligible performance deterioration (less than 5%) when comparedto symmetric encryption schemes, and considerablyimproves the performance of existing AON encryptionschemes.
- We propose Bastion, an efficient scheme whichensures data confidentiality against an adversarythat knows the encryption key and has access toa large fraction of the ciphertext blocks.
- We analyze the security of Bastion, and we showthat it prevents leakage of any plaintext block aslong as the adversary has access to the encryptionkey and to all but two ciphertext blocks.
- We evaluate the performance of Bastion analyticallyand empirically in comparison to anumber of existing encryption techniques. Ourresults show that Bastion considerably improves(by more than 50%) the performance of existingAON encryption schemes, and only incurs anegligible overhead when compared to existingsemantically secure encryption modes (e.g., theCTR encryption mode).
- We discuss practical insights with respect to thedeployment of Bastion within existing storagesystems, such as the HYDRAstor grid storagesystem.

SYSTEM ARCHITECTURE:



SYSTEM REQUIREMENTS:

HARDWARE REQUIREMENTS:

- System : Pentium Dual Core.
- Hard Disk : 120 GB.
- Monitor : 15'' LED
- Input Devices : Keyboard, Mouse
- Ram : 1 GB

SOFTWARE REQUIREMENTS:

- Operating system : Windows 7.
- Coding Language : JAVA/J2EE
- Tool : Netbeans 7.2.1
- Database : MYSQL

REFERENCE:

Ghassan O. Karame, *Member, IEEE*, Claudio Soriente, *Member, IEEE*, Krzysztof Lichota, SrdjanCapkun, *Senior Member, IEEE*, "Securing Cloud Data under Key Exposure", **IEEE Transactions on Cloud Computing, 2017.**