**LITERATURE SURVEY**

**2.1 Publicly verifiable secure cloud storage for dynamic data using secure network coding**

**AUTHORS: B. Sengupta and S. Ruj**

**ABSTRACT:** Cloud service providers offer storage outsourcing facility to their clients. In a secure cloud storage (SCS) protocol, the integrity of the client's data is maintained. In this work, we construct a publicly verifiable secure cloud storage protocol based on a secure network coding (SNC) protocol where the client can update the outsourced data as needed. To the best of our knowledge, our scheme is the first SNC-based SCS protocol for dynamic data that is secure in the standard model and provides privacy-preserving audits in a publicly verifiable setting. Furthermore, we discuss, in details, about the (im)possibility of providing a general construction of an efficient SCS protocol for dynamic data (DSCS protocol) from an arbitrary SNC protocol. In addition, we modify an existing DSCS scheme (DPDP I) in order to support privacy-preserving audits. We also compare our DSCS protocol with other SCS schemes (including the modified DPDP I scheme). Finally, we figure out some limitations of an SCS scheme constructed using an SNC protocol.

**2.2** **Enabling public auditability and data dynamics for storage security in cloud computing**

**AUTHORS: Q. Wang, C. Wang, K. Ren, W. Lou, and J. Li**

**ABSTRACT:** Cloud Computing has been envisioned as the next-generation architecture of IT Enterprise. It moves the application software and databases to the centralized large data centers, where the management of the data and services may not be fully trustworthy. This unique paradigm brings about many new security challenges, which have not been well understood. This work studies the problem of ensuring the integrity of data storage in Cloud Computing. In particular, we consider the task of allowing a third party auditor (TPA), on behalf of the cloud client, to verify the integrity of the dynamic data stored in the cloud. The introduction of TPA eliminates the involvement of the client through the auditing of whether his data stored in the cloud are indeed intact, which can be important in achieving economies of scale for Cloud Computing. The support for data dynamics via the most general forms of data operation, such as block modification, insertion, and deletion, is also a significant step toward practicality, since services in Cloud Computing are not limited to archive or backup data only. While prior works on ensuring remote data integrity often lacks the support of either public auditability or dynamic data operations, this paper achieves both. We first identify the difficulties and potential security problems of direct extensions with fully dynamic data updates from prior works and then show how to construct an elegant verification scheme for the seamless integration of these two salient features in our protocol design. In particular, to achieve efficient data dynamics, we improve the existing proof of storage models by manipulating the classic Merkle Hash Tree construction for block tag authentication. To support efficient handling of multiple auditing tasks, we further explore the technique of bilinear aggregate signature to extend our main result into a multiuser setting, where TPA can perform multiple auditing tasks simultaneously. Extensive security and performance analysis show that the proposed schemes are highly efficient and provably secure.

**2.3 Network information flow**

**AUTHORS: R. Ahlswede, N. Cai, S. R. Li, and R. W. Yeung**

**ABSTRACT:** We introduce a new class of problems called network information flow which is inspired by computer network applications. Consider a point-to-point communication network on which a number of information sources are to be multicast to certain sets of destinations. We assume that the information sources are mutually independent. The problem is to characterize the admissible coding rate region. This model subsumes all previously studied models along the same line. We study the problem with one information source, and we have obtained a simple characterization of the admissible coding rate region. Our result can be regarded as the max-flow min-cut theorem for network information flow. Contrary to one's intuition, our work reveals that it is in general not optimal to regard the information to be multicast as a "fluid" which can simply be routed or replicated. Rather, by employing coding at the nodes, which we refer to as network coding, bandwidth can in general be saved. This finding may have significant impact on future design of switching systems.

# 2.4 Linear network coding

**AUTHORS: S. R. Li, R. W. Yeung, and N. Cai**

**ABSTRACT:** Consider a communication network in which certain source nodes multicast information to other nodes on the network in the multihop fashion where every node can pass on any of its received data to others. We are interested in how fast each node can receive the complete information, or equivalently, what the information rate arriving at each node is. Allowing a node to encode its received data before passing it on, the question involves optimization of the multicast mechanisms at the nodes. Among the simplest coding schemes is linear coding, which regards a block of data as a vector over a certain base field and allows a node to apply a linear transformation to a vector before passing it on. We formulate this multicast problem and prove that linear coding suffices to achieve the optimum, which is the max-flow from the source to each receiving node.

**2.5**  **Homomorphic MACs: MAC-based integrity for network coding**

**AUTHORS: S. Agrawal and D. Boneh**

**ABSTRACT:** Network coding has been shown to improve the capacity and robustness in networks. However, since intermediate nodes modify packets en-route, integrity of data cannot be checked using traditional MACs and checksums. In addition, network coded systems are vulnerable to pollution attacks where a single malicious node can flood the network with bad packets and prevent the receiver from decoding the packets correctly. Signature schemes have been proposed to thwart such attacks, but they tend to be too slow for online per-packet integrity. Here we propose a homomorphic MAC which allows checking the integrity of network coded data. Our homomorphic MAC is designed as a drop-in replacement for traditional MACs (such as HMAC) in systems using network coding.