RSNI3

Rancangan Standar Nasional Indonesia 3

RSNI3 ISO/IEC 19592-1:2016
(Ditetapkan oleh BSN tahun 2024)

Teknologi informasi — Teknik keamanan — *Secret sharing* — Bagian 1: Umum

Information technology — Security techniques — Secret sharing — Part 1: General

(ISO/IEC 19592-1: 2016, IDT)

ICS 35.030

BADAN STANDARDISASI NASIONAL

BSN
Email: dokinfo@bsn.go.id
www.bsn.go.id

Diterbitkan di Jakarta

Daftar isi

Prakata

SNI ISO/IEC 19592-1:2016 *Teknologi informasi — Teknik keamanan — Secret sharing — Bagian* 1: *Umum*, merupakan standar yang disusun dengan jalur adopsi tingkat keselarasan identik dari ISO/IEC 19592-1:2016 *Information technology — Security techniques — Secret sharing — Part* 1: *General*, dengan metode adopsi terjemahan dua bahasa dan ditetapkan oleh BSN Tahun 2024.

Standar ini disusun oleh Komite Teknis 35-04, Keamanan Informasi, Keamanan Siber, dan Perlindungan Privasi. Standar ini telah dibahas dan disepakati dalam rapat konsensus pada tanggal 21 Juni 2024 di Depok, yang dihadiri oleh para pemangku kepentingan (*stakeholders*) terkait yaitu perwakilan dari pemerintah, pelaku usaha, konsumen, dan pakar. Standar ini telah melalui tahap jajak pendapat pada tanggal 18 Juli 2024 sampai dengan 1 Agustus 2024 dengan hasil akhir disetujui menjadi SNI.

Kosakata yang digunakan dalam Standar ini mengikuti bentuk baku yang dicantumkan dalam Kamus Besar Bahasa Indonesia (KBBI), tetapi ada beberapa kosakata yang belum ada di dalam KBBI.

Kata/istilah "*secret sharing", "share", "dealer", "superset", "homomorphicity", "cleartext*" tidak diterjemahkan dalam Standar ini karena Komite Teknis 35-04 belum menemukan padanan kata/istilah yang sesuai dengan konteks dalam Bahasa Indonesia.

Apabila pengguna menemukan keraguan dalam Standar ini, disarankan untuk melihat standar aslinya yaitu ISO/IEC 19592-1:2016 dan/atau dokumen terkait lain yang menyertainya.

Perlu diperhatikan bahwa kemungkinan beberapa unsur dari Standar ini dapat berupa hak kekayaan intelektual (HAKI). Namun selama proses perumusan SNI, Badan Standardisasi Nasional telah memperhatikan penyelesaian terhadap kemungkinan adanya HAKI terkait substansi SNI. Apabila setelah penetapan SNI masih terdapat permasalahan terkait HAKI, Badan Standardisasi Nasional tidak bertanggung jawab mengenai bukti, validitas, dan ruang lingkup dari HAKI tersebut.

Pendahuluan

Skema *secret sharing* adalah teknik kriptografis yang digunakan untuk memproteksi konfidensialitas suatu pesan dengan membaginya menjadi beberapa bagian yang disebut *shares*. Sebuah skema *secret sharing* memiliki dua bagian utama: algoritma berbagi pesan untuk membagi pesan menjadi beberapa *shares* dan algoritma rekonstruksi pesan untuk memulihkan pesan dari semua atau sebagian dari *shares*.

Secret sharing dapat digunakan untuk menyimpan data (misalnya, nilai konfidensial atau kunci kriptografis) dengan aman dalam sistem terdistribusi. Selain itu, *secret sharing* adalah teknologi fundamental untuk komputasi multi-pihak yang aman yang dapat digunakan untuk memproteksi pemrosesan data dalam suatu sistem terdistribusi. Untuk memfasilitasi penggunaan teknologi secara efektif dan memelihara interoperabilitas, ISO/IEC 19592 (seluruh bagian) menspesifikasikan *secret sharing* dan teknologi terkait.

Teknologi informasi — Teknik keamanan — *Secret sharing* — Bagian 1: Umum

1 Ruang lingkup

ISO/IEC 19592 (semua bagian) menspesifikasikan skema *secret sharing* kriptografis dan propertinya. Dokumen ini mendefinisikan para pihak yang terlibat dalam skema *secret sharing*, terminologi yang digunakan dalam konteks skema *secret sharing*, parameter dan properti skema tersebut.

2 Acuan normatif

Tidak ada acuan normatif dalam dokumen ini

3 Istilah dan definisi

Untuk tujuan dokumen ini, istilah dan definisi berikut berlaku.

ISO dan IEC memelihara basis data terminologi untuk penggunaan dalam standardisasi di alamat berikut:

— IEC Electropedia: tersedia di http://www.electropedia.org/

— ISO *Online browsing platform* : tersedia di http://www.iso.org/obp

3.1
struktur akses
himpunan dari himpunan bagian dari seluruh pemegang-*share* (3.11), $\boldsymbol{A} \subset \{S \mid S \subset \{1,\ldots,n\}\}$, sedemikian sehingga untuk semua $S,T \in \boldsymbol{A}$, S bukan merupakan himpunan bagian dari T dan T bukan merupakan himpunan bagian dari S dan *shares* (3.10) yang dipegang oleh pemegang-*share* dalam S cukup untuk berhasil merekonstruksi pesan tersebut (3.4) menggunakan algoritma rekonstruksi pesan (3.5)

3.2
struktur lawan
himpunan dari himpunan bagian dari seluruh pemegang-*share*(3.11), $\boldsymbol{D} \subset \{S \mid S \subset \{1,\ldots,n\}\}$, sedemikian sehingga untuk semua $S,T \in \boldsymbol{D}$, S bukan himpunan bagian dari T dan T bukan himpunan bagian dari S dan adalah tidak mungkin untuk merekonstruksi pesan (3.4) dari *shares* (3.10) yang dimiliki oleh pemegang-*share* di dalam S

3.3
dealer
pihak yang menjalankan algoritma berbagi pesan (3.6)

3.4
pesan
informasi rahasia yang harus diproteksi
CONTOH Suatu nilai konfidensial atau kunci kriptografis

3.5
algoritma rekonstruksi pesan
proses yang mentransformasi suatu himpunan bagian yang dapat dipulihkan dari elemen-elemen di dalam suatu vektor *share* (3.13) menjadi pesan asli (3.4)

3.6
algoritma berbagi pesan
proses yang mengubah pesan (3.4) menjadi vektor *share* (3.13)

3.7
ruang pesan
himpunan pesan (3.4) yang dapat dibagikan oleh suatu skema *secret sharing* (3.9)

3.8
penerima
pihak yang menjalankan algoritma rekonstruksi pesan (3.5)

3.9
skema *secret sharing*
teknik kriptografis yang digunakan untuk memproteksi konfidensialitas suatu pesan (3.4) dengan membaginya menjadi sejumlah keping yang disebut *shares* (3.10)

CATATAN 1 untuk entri: Skema tersebut terdiri dari dua proses komponen: algoritma berbagi pesan dan algoritma rekonstruksi pesan.

3.10
share
elemen vektor *share* (3.13)

3.11
pemegang *share*
pihak yang menyimpan sebuah *share* keluaran (oleh) algoritma berbagi pesan (3.6))

3.12
ruang *share*
himpunan elemen yang dapat berada di vektor *share* (3.13) dari skema *secret sharing* (3.9)

3.13
vektor *share*
vektor dari nilai-nilai yang dikeluarkan oleh algoritma berbagi pesan (3.6)

3.14
ambang (*threshold*)
jumlah minimal elemen yang tidak dimodifikasi dalam vektor *share* (3.13) yang dibutuhkan agar berhasil merekonstruksi pesan (3.4)

4 Model umum *secret sharing*

4.1 Para pihak yang terlibat

Pengoperasian skema *secret sharing* melibatkan tiga peran berikut:

a) *dealer*;

b) pemegang *share*

c) penerima.

Dealer adalah pihak yang mempunyai pesan dan menjalankan algoritma berbagi pesan. Setelah menjalankan algoritma pada pesan tersebut untuk mendapatkan vektor *share*, *dealer* mendistribusikan *share* dalam vektor *share* tersebut kepada para pemegang *share*. Cara pendistribusian *shares* kepada para pemegang *share* berupa aplikasi-spesifik dan berada di luar cakupan ISO/IEC 19592 (semua bagian).

Penerima adalah pihak yang berupaya merekonstruksi pesan tersebut. Ketika penerima ingin mempelajari pesan tersebut, penerima mengumpulkan *shares* dari sekumpulan pihak yang berwenang dan merangkai vektor *share* untuk diteruskan ke algoritma rekonstruksi pesan. Jika tersedia cukup *shares* untuk merekonstruksi pesan, penerima mempelajari pesan tersebut dengan menjalankan algoritma rekonstruksi pesan. Penerima boleh mengumpulkan *shares* tambahan untuk meningkatkan peluang keberhasilan rekonstruksi. Cara pengumpulan *shares* dari para pemegang *share* berupa aplikasi-spesifik dan berada di luar cakupan ISO/IEC 19592 (semua bagian).

Satu pihak dapat mempunyai lebih dari satu peran. Sebagai contoh, di antara sejumlah pihak, masing-masing pihak boleh mempunyai pesan yang ingin dibagikan kepada semua pihak, termasuk dirinya sendiri. Dalam skenario seperti itu, masing-masing pihak merupakan *dealer* sekaligus pemegang *share*

4.2 Parameter

4.2.1 Ikhtisar

Parameter berikut ini berlaku untuk semua skema *secret sharing* yang dispesifikasikan dalam ISO/IEC 19592 (semua bagian):

a) ruang pesan, dijelaskan dalam 4.2.2;

b) ruang *share*, dijelaskan dalam 4.2.3;

c) jumlah *share*, dijelaskan dalam 4.2.4;

d) struktur akses, dijelaskan dalam 4.2.5.

4.2.2 Ruang pesan

Ruang pesan adalah himpunan nilai yang mungkin untuk pesan tersebut, yaitu rahasia yang akan dibagi menjadi beberapa *shares* oleh algoritma berbagi pesan. Selagi skema *secret sharing* boleh mengizinkan berbagai kemungkinan rentang ruang pesan yang mungkin (misalnya untuk tipe data yang berbeda) dalam sebarang instansiasi spesifik apapun, ruang pesan harus tetap, dan semua pengguna skema harus mengetahui detail-detail ruang pesan tersebut.

4.2.3 Ruang *share*

Ruang *share* adalah himpunan elemen dari mana *shares* pesan diseleksi. Algoritma berbagi pesan mengeluarkan vektor *share* yang berisi elemen dari ruang *share*. Bagi banyak skema *secret sharing*, pilihan ruang pesan secara langsung menetapkan ruang *share*.

4.2.4 Jumlah *share*

Skema *secret sharing* biasanya mampu membagi pesan input menjadi sejumlah berhingga *shares*. Dalam praktiknya, skema yang membagi pesan menjadi dua atau lebih *shares* disyaratkan. Setiap instansiasi skema *secret sharing* mendefinisikan algoritma berbagi pesan yang mengeluarkan vektor *share* yang berisi n *shares*.

Serupa halnya, instansiasi skema *secret sharing* mendefinisikan algoritma rekonstruksi pesan yang menerima vektor *share* dengan jumlah elemen yang tetap ini. Perhatikan bahwa beberapa skema *secret sharing* dapat merekonstruksi pesan bahkan ketika beberapa nilai dalam vektor *share* dimodifikasi atau hilang.

CATATAN instansiasi skema *secret sharing* sering kali menetapkan rentang jumlah *share* yang mungkin dengan batas atas dan bawah. Sebagai contoh, algoritma berbagi pesan dapat diimplementasikan sehingga selalu mengeluarkan suatu vektor *share* dengan n *share*, namun bergantung pada aplikasinya, algoritma tersebut juga dapat mengeluarkan t *share* dengan $2 \leq t \leq n$.

4.2.5 Struktur akses

Pengoperasian skema *secret sharing* secara fundamental bergantung pada struktur akses yang terkait. Struktur akses adalah kumpulan minimal dari himpunan bagian *shares* yang mungkin, yang diperlukan sebagai input agar algoritma rekonstruksi pesan berhasil mengeluarkan pesan. Artinya, dengan diberikannya koleksi *shares*, maka dapat digunakan untuk merekonstruksi pesan jika dan hanya jika koleksi *share* tersebut berisi satu atau lebih dari himpunan bagian *share* di dalam struktur akses.

Beberapa skema memiliki ambang batas terkait – jumlah *shares* yang benar yang harus diberikan ke algoritma rekonstruksi pesan agar berhasil merekonstruksi pesan. Sebagai contoh, jika skema *secret sharing* mendukung ambang batas, skema tersebut boleh dipakai untuk membagi pesan kedalam n *shares* dengan ambang batas, k, dengan $2 \leq k \leq n$. Dalam pengaturan seperti itu, sebarang k *shares* cukup untuk keberhasilan penyelesaian algoritma rekonstruksi pesan. Artinya, struktur akses terdiri dari semua himpunan bagian dari k *share*, yaitu semua himpunan bagian dari kardinalitas k.

Suatu skema *secret sharing* juga dapat diinstansiasi dengan struktur akses sesuai pesanan yang berisi himpunan pihak yang dapat merekonstruksi pesan dengan menggabungkan *share* mereka. Sebagai contoh, untuk empat pemegang *share*, struktur akses dapat menspesifikasikan bahwa *share* m_1, m_2, m_3 cukup untuk merekonstruksi rahasia begitu pula dengan m_1 dan m_4 atau m_2 dan m_4, sehingga menghasilkan struktur akses $\mathbf{A} = \{\{1, 2, 3\}, \{1, 4\}, \{2, 4\}\}$. Dalam hal kasus ini, pihak 3 dan 4 atau 1 dan 2, sebagai contoh, tidak dapat memulihkan rahasianya sendiri, namun semua himpunan pihak di $\mathbf{A}$, dan juga *superset*-nya, dapat merekonstruksi pesan tersebut. Untuk contoh ini, struktur lawan dalam kasus ini adalah $\mathbf{D} = \{\{1, 2\}, \{1, 3\}, \{2, 3\}, \{3, 4\}\}$.

4.3 Proses berbagi pesan

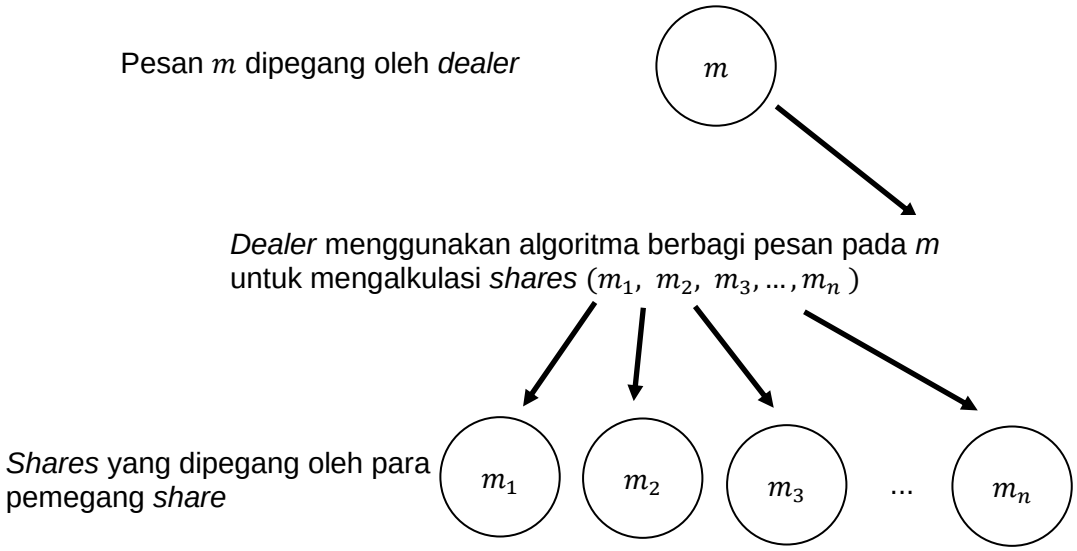
Proses berbagi pesan terdiri dari tiga langkah berikut.

a) *Dealer* menjalankan algoritma berbagi pesan pada pesan, m, dan memperoleh vektor *share* $(m_1, m_2, \ldots, m_n)$.

b) *Dealer* mendistribusikan elemen-elemen dalam vektor *share* kepada para pemegang *share*

c) Para pemegang *share* menyimpan *share* tersebut dengan cara yang aman.

Gambar 1 mengilustrasikan contoh dari suatu proses berbagi pesan.



Gambar 1 — Contoh suatu proses *secret sharing*

Penerap skema *secret sharing* sebaiknya mempertimbangkan untuk menghapus salinan *share* milik *dealer* setelah pendistribusian *share*, kecuali hal ini dicegah oleh persyaratan aplikasi.

4.4 Proses rekonstruksi pesan

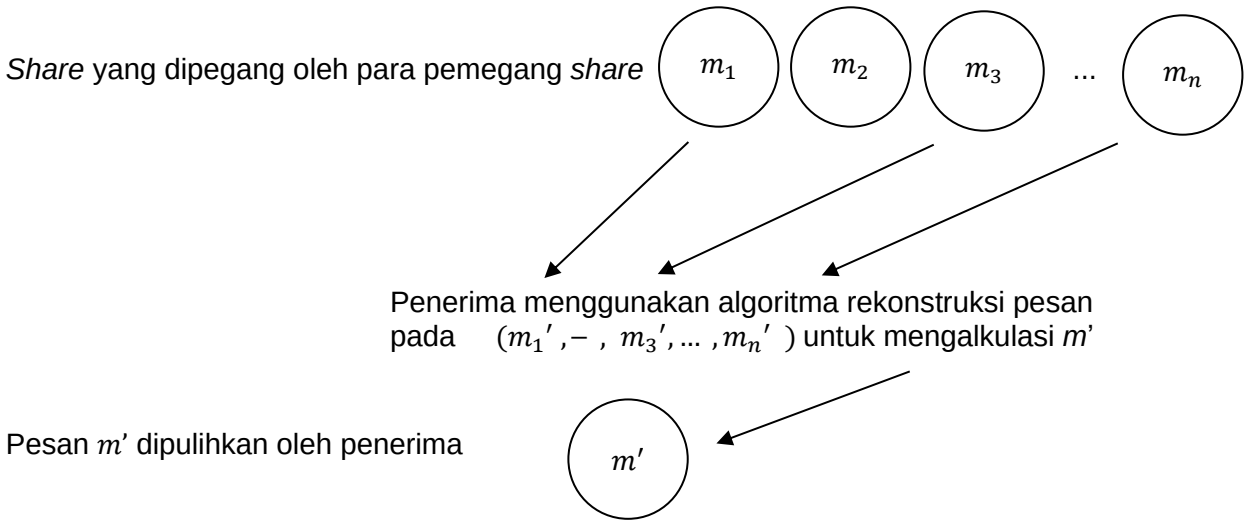
Proses rekonstruksi pesan terdiri dari tiga langkah berikut.

a) Suatu himpunan bagian dari para pemegang *share* mengirimkan *share* pesan mereka, m, ke penerima.

b) Penerima menjalankan algoritma rekonstruksi pesan pada versi *shares* yang diterima dalam upaya mempelajari pesan tersebut. Jika algoritma berhasil, penerima memulihkan m', yang akan sama dengan pesan asli, m, jika *shares* yang diterima semuanya benar. Namun, rekonstruksi juga mungkin gagal. Dalam kasus itu, penerima tidak mempelajari apa pun tentang pesan m tersebut, kecuali, mungkin, panjangnya pesan.

Gambar 2 mengilustrasikan contoh proses rekonstruksi pesan. Dalam contoh ini, penerima tidak menerima versi m_2' dari *share* asli m_2. Contoh ini juga mengasumsikan bahwa skema *secret sharing* yang digunakan tidak mensyaratkan m_2' untuk merekonstruksi m.

CATATAN Dalam praktiknya, pemegang *share* perusak mungkin juga mencoba mendampak rekonstruksi dengan mengirimkan nilai m_i' yang tidak sama dengan m_i. Beberapa skema *secret sharing* mampu mendeteksi jika satu atau lebih dari para pemegang *share* mengirimkan *share* palsu



Gambar 2 — Contoh suatu proses rekonstruksi pesan

5 Properti skema berbagi pesan

5.1 Persyaratan fundamental

5.1.1 Ikhtisar

Dua persyaratan fundamental yang harus dipenuhi oleh skema *secret sharing* dalam ISO/IEC 19592 (seluruh bagian) adalah konfidensialitas pesan dan pemulihan pesan. Persyaratan konfidensialitas pesan memastikan kerahasiaan pesan yang dibagikan dan persyaratan pemulihan pesan memastikan ketersediaan.

5.1.2 Konfidensialitas pesan

Konfidensialitas pesan untuk skema *secret sharing* berarti bahwa jika himpunan pemegang *share* mempunyai himpunan bagian *shares* yang tidak mencukupi untuk rekonstruksi, maka tidaklah mungkin bagi para pemegang *share* tersebut untuk mempelajari apa pun tentang pesan tersebut (selain, mungkin, panjang pesan).

CATATAN Sebagai contoh, pertimbangkan seorang pengguna yang menggunakan skema *secret sharing* untuk membagi file konfidensial ke dalam sejumlah potongan dan mengunggah setiap potongan ke server terpisah. Skema *secret sharing* yang dibuat oleh pengguna memastikan bahwa semua potongan file diperlukan untuk merekonstruksinya. Ini berarti bahwa hanya satu kumpulan server lengkap yang diotorisasi untuk memulihkan pesan, dan tidak ada himpunan bagian server yang tidak lengkap yang dapat mempelajari apa pun tentang file konfidensial tersebut dari *share* yang mereka miliki.

5.1.3 Pemulihan pesan

Pemulihan pesan untuk skema *secret sharing* berarti bahwa jika suatu himpunan pemegang *share* mempunyai suatu himpunan bagian *shares* yang cukup untuk rekonstruksi, para pemegang *share* ini dapat merekonstruksi pesan dengan menggabungkan *shares* mereka dan menerapkan algoritma rekonstruksi pesan.

CATATAN Sebagai contoh, pertimbangkan sebuah perusahaan yang memproteksi kunci privat yang digunakan untuk membuat tanda tangan menggunakan suatu skema *secret sharing*. Skema secret sharing diatur sedemikian rupa sehingga memperbolehkan presiden perusahaan merekonstruksi kunci rahasia sendirian dari *shares* yang dimilikinya. Selain itu, setidaknya diperlukan dua wakil presiden untuk merekonstruksi kunci dan membuat tanda tangan yang valid. Dalam hal ini, presiden diotorisasi dan setiap himpunan bagian dari dua wakil presiden diotorisasi.

5.2 Persyaratan opsional

5.2.1 Ikhtisar

Terlepas dari persyaratan fundamental di atas, skema *secret sharing* boleh menunjukkan properti lain yang berkaitan dengan parameter dan prosesnya. Contoh properti tersebut dijelaskan dalam 5.2.2 dan 5.2.3.

5.2.2 *Homomorphicity*

Skema *secret sharing* dapat mendukung satu atau lebih operasi *homomorphicity* pada himpunan *share* yang mungkin. Pada prinsipnya, operasi *homomorphicity* membolehkan *vektor share* untuk digabungkan menjadi sebuah *vektor share* baru dalam suatu cara sehingga *vektor share* baru merepresentasikan hasil operasi yang bermakna pada pesan asli atau pesan-pesan.

Pertimbangkan skema *secret sharing* n-pihak dengan ruang pesan M, ruang *share* S, algoritma berbagi pesan *Share* dan algoritma rekonstruksi pesan $Reconstruct$. Kemudian, misalkan $\oplus$ adalah operasi yang terdefinisi pada elemen M dan misalkan $\otimes$ adalah operasi yang terdefinisi pada vektor-vektor *share* dalam himpunan S^n. Skema *secret sharing* ini adalah ($\oplus$,$\otimes$)-*homomorphicity* jika, untuk semua pesan, $m_1, m_2 \in M$,

$$Reconstruct\,[Share\,(m_1)\ \otimes\ Share\,(m_2)]\ =\ m_1 \oplus m_2$$

Operasi *homomorphicity* seperti itu berguna dalam aplikasi seperti komputasi aman, tempat peranti mampu melakukan operasi komputasi yang dispesifikasikan pada data tanpa memiliki akses ke nilai *cleartext* dari data tersebut.

5.2.3 Kemampuan verifikasi (*Verifiability*)

Skema *secret sharing* dapat dilakukan verifikasi jika *shares* yang dikeluarkan oleh algoritma berbagi pesan berisi informasi tambahan yang memperbolehkan para pemegang *share* untuk memverifikasi bahwa *shares* tersebut benar, yaitu membolehkan keberhasilan rekonstruksi. Sebelum rekonstruksi, baik para pemegang *share* maupun penerima dapat menggunakan informasi tambahan ini untuk mengecek kebenaran *shares* tersebut. Metode lain untuk memverifikasi bahwa suatu skema berbagi konsisten adalah dengan menambahkan informasi tambahan pada *shares* yang dapat digunakan untuk memverifikasi kebenaran dari pesan yang direkonstruksi tetapi bukan *shares* yang digunakan dalam proses rekonstruksi.

5.3 Properti lain

5.3.1 Ikhtisar

Berbagai karakteristik operasional dan keamanan skema *secret sharing* dapat diukur, termasuk yang dijelaskan dalam 5.3.2 hingga 5.3.4.

5.3.2 Jaminan konfidensialitas

Properti konfidensialitas pesan untuk skema *secret sharing* dapat berbasiskan pada berbagai asumsi. Sebagai contoh, skema *secret sharing* adalah bersifat konfidensial secara teori informasi jika skema tersebut menjaga konfidensialitas pesan tanpa memedulikan kekuatan komputasi dari koalisi tanpa otorisasi para pemegang *share* yang mencoba mengakses pesan tersebut. Serupa halnya, skema *secret sharing* bersifat konfidensial secara komputasi jika skema tersebut menjamin konfidensialitas terhadap koalisi tanpa otorisasi dengan kemampuan komputasi terbatas.

Selanjutnya, beberapa tingkat konfidensialitas berdasarkan teori informasi dapat dibedakan. Konfidensialitas sempurna ketika sama sekali tidak ada informasi yang diungkapkan tentang pesan tersebut. Namun, skema *secret sharing* mungkin hanya memberikan tingkat konfidensialitas yang lebih rendah, yang berarti bahwa koalisi tanpa otorisasi dari para pihak dapat mempelajari sejumlah tidak-nol dari bit pesan dengan menggabungkan *shares* mereka dengan suatu cara tertentu.

5.3.3 Kompleksitas

Kompleksitas suatu algoritma dalam skema *secret sharing* adalah jumlah unit operasi yang disyaratkan untuk mengeksekusi algoritma tersebut. Kompleksitasnya dispesifikasikan secara terpisah untuk setiap algoritma yang terkait dengan skema *secret sharing*.

5.3.4 Laju informasi

Laju informasi adalah properti skema *secret sharing* yang merepresentasikan rasio ukuran pesan terhadap ukuran *share* maksimum yang mungkin untuk pesan tertentu. Jika pesan dan *share* mempunyai ukuran yang sama, laju informasi adalah 1 dan skema *secret sharing* disebut ideal.

Terdapat skema *secret sharing* yang aman secara komputasi rahasia, sebagai contoh, berbagai skema *ramp,* yang rasionya dapat lebih besar dari satu, karena pesannya lebih besar dari suatu *share* tunggal.

Information technology — Security techniques — Secret sharing — Part 1: General

1 Scope

ISO/IEC 19592 (all parts) specifies cryptographic secret sharing schemes and their properties. This document defines the parties involved in a secret sharing scheme, the terminology used in the context of secret sharing schemes, the parameters and the properties of such a scheme.

2 Normative references

There are no normative references in this document.

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

— IEC Electropedia: available at http://www.electropedia.org/

— ISO Online browsing platform: available at http://www.iso.org/obp

3.1
access structure
set of subsets of all *share-holders* (3.11), $\boldsymbol{A} \subset \{S \mid S \subset \{1, \ldots, n\}\}$, such that for all $S, T \in \boldsymbol{A}$, S is not a subset of T and T is not a subset of S and the *shares* (3.10) held by share-holders in S are sufficient to successfully reconstruct the *message* (3.4) using the *message reconstruction algorithm* (3.5)

3.2
adversary structure
set of subsets of all *share-holders* (3.11), $\boldsymbol{D} \subset \{S \mid S \subset \{1, \ldots, n\}\}$, such that for all $S, T \in \boldsymbol{D}$, S is not a subset of T and T is not a subset of S and it is not possible to reconstruct the *message* (3.4) from the *shares* (3.10) held by share-holders in S

3.3
dealer
party running the *message sharing algorithm* (3.6)

3.4
message
secret information that is to be protected
EXAMPLE A confidential value or cryptographic key.

3.5
message reconstruction algorithm
process which transforms a recoverable subset of elements in a *share vector* (3.13) into the original *message* (3.4)

3.6
message sharing algorithm
process which transforms *messages* (3.4) into a *share vector* (3.13)

3.7
message space
set of *messages* (3.4) that can be shared by a *secret sharing scheme* (3.9)

3.8
receiver
party running the *message reconstruction algorithm* (3.5)

3.9
secret sharing scheme
cryptographic technique used to protect the confidentiality of a *message* (3.4) by dividing it into a number of pieces called *shares* (3.10)
Note 1 to entry: It consists of two component processes: a message sharing algorithm and a message reconstruction algorithm.

3.10
share
element of the *share vector* (3.13)

3.11
share-holder
party storing a share output by the *message sharing algorithm* (3.6)

3.12
share space
set of elements that can occur in a *share vector* (3.13) of a *secret sharing scheme* (3.9)
3.13
share vector
vector of values output by the *message sharing algorithm* (3.6)

3.14
threshold
minimal number of unmodified elements in the *share vector* (3.13) that are needed to successfully reconstruct the *message* (3.4)

4 General model of secret sharing

4.1 Parties involved

The operation of a secret sharing scheme involves the following three roles:

a) the dealer;

b) the share-holder;

c) the receiver.

The dealer is the party that has a message and runs the message sharing algorithm. After running the algorithm on the message to obtain the share vector, it distributes the shares in the share vector to the share-holders. The way in which shares are distributed to share-holders is application-specific and is outside the scope of ISO/IEC 19592 (all parts).

The receiver is the party that attempts to reconstruct the message. When the receiver wants to learn the message, it collects shares from an authorized set of parties and assembles a share vector to pass to the message reconstruction algorithm. If enough shares are available to reconstruct the message, the receiver learns the message by running the message reconstruction algorithm. The receiver may collect additional shares to increase its chances of successful reconstruction. The way in which shares are collected from share-holders is application-specific and is outside the scope of ISO/IEC 19592 (all parts).

A party can have more than one role. For example, among a number of parties, each may have a message that it wants to share among all the parties, including itself. In such a scenario, each party is both a dealer and share-holder.

4.2 Parameters

4.2.1 Overview

The following parameters apply to all secret sharing schemes specified in ISO/IEC 19592 (all parts):

a) the message space, described in 4.2.2;

b) the share space, described in 4.2.3;

c) the number of shares, described in 4.2.4;

d) the access structure, described in 4.2.5.

4.2.2 Message space

The message space is the set of possible values for the message, i.e. the secret that is to be divided into shares by the message sharing algorithm. Whilst a secret sharing scheme might permit a range of possible message spaces (e.g. for different data types) in any specific instantiation, the message space shall be fixed, and all users of the scheme shall know the details of the message space.

4.2.3 Share space

The share space is the set of elements that the shares of a message are selected from. The message sharing algorithm outputs a share vector that contains elements from the share space. For many secret sharing schemes, the choice of message space directly fixes the share space.

4.2.4 Number of shares

A secret sharing scheme is typically able to divide an input message into any finite number of shares. In practice, schemes that divide a message into two or more shares are required. Each instantiation of a secret sharing scheme defines a message sharing algorithm that outputs a share vector containing n shares.

Similarly, the instantiation of the secret sharing scheme defines a message reconstruction algorithm that accepts a share vector with this fixed number of elements. Note that some secret sharing schemes can reconstruct the message even when some values in the share vector are modified or missing.

NOTE An instantiation of a secret sharing scheme often fixes a range for the possible number of shares with upper and lower bounds. For example, a message sharing algorithm can be implemented so that

it always outputs a share vector with n shares but, depending on the application, it could also output t shares where $2 \leq t \leq n$.

4.2.5 Access structure

The operation of a secret sharing scheme is fundamentally dependent on its associated access structure. An access structure is the minimal set of possible subsets of shares that are needed as input in order for the message reconstruction algorithm to successfully output the message. That is, given a collection of shares, it can be used to reconstruct the message if and only if it contains one or more of the share subsets in the access structure.

Some schemes have an associated threshold – the number of correct shares that have to be provided to the message reconstruction algorithm in order for it to successfully reconstruct the message. For example, if a secret sharing scheme supports thresholds, it might be instantiated to share the message into *n* shares with a threshold, k, where $2 \leq \mathrm{k} \leq \mathrm{n}$. In such a setting, any k shares are sufficient for a successful completion of the message reconstruction algorithm. That is, the access structure consists of all k-subsets of shares, i.e. all subsets of cardinality k.

A secret sharing scheme can also be instantiated with a custom access structure containing sets of parties who can reconstruct the message by combining their shares. For example, for four share-holders, an access structure can specify that shares m_1, m_2, m_3 are sufficient for reconstructing the secret as well as m_1 and m_4 or m_2 and m_4, resulting in an access structure $\boldsymbol{A} = \{\{1,2,3\},\{1,4\},\{2,4\}\}$. In this case, parties 3 and 4 or 1 and 2, for example, cannot restore the secret on their own, but all sets of parties in $\boldsymbol{A}$, as well as their supersets, can reconstruct the message. For this example, the adversary structure in this case is $D = \{\{1,2\},\{1,3\},\{2,3\},\{3,4\}\}$.

4.3 Message sharing process

The message sharing process consists of the following three steps.

a) The dealer runs the message sharing algorithm on the message, m, and obtains the share vector $(m_1, m_2, \ldots, m_n)$.

b) The dealer distributes the elements in the share vector to the share-holders.

c) The share-holders store the shares in a secure way.

Figure 1 illustrates an example of a message sharing process.

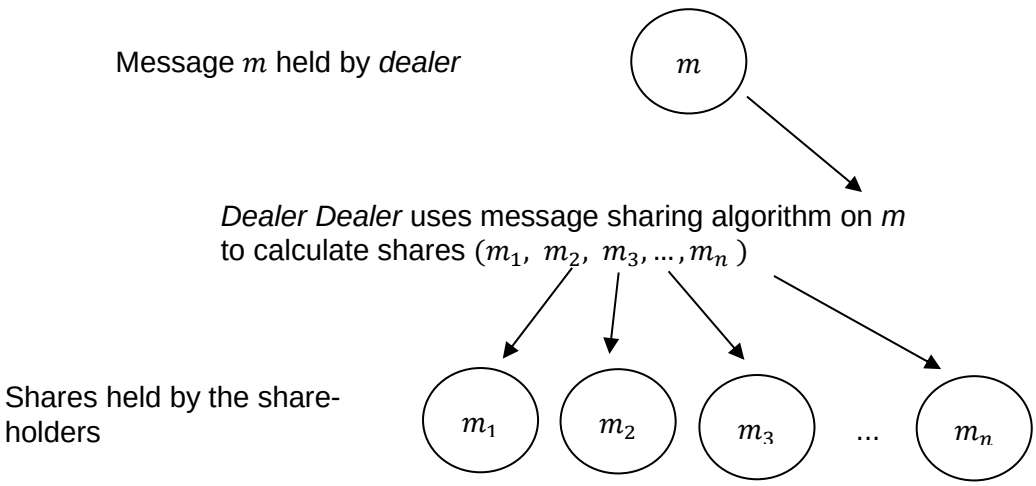


Figure 1 — Example of a secret sharing process

The implementer of a secret sharing scheme should consider erasing the dealer's copies of the shares after share distribution, unless this is prevented by the application requirements.

4.4 Message reconstruction process

The message reconstruction process consists of the following three steps.

a) A subset of the share-holders send their shares of the message, m, to the receiver.

b) The receiver runs the message reconstruction algorithm on the received versions of the shares in an attempt to learn the message. If the algorithm succeeds, the receiver recovers m', which will equal the original message, m, if the received shares were all correct. However, the reconstruction may also fail. In that case, the receiver learns nothing about the message, m, except, perhaps, its length.

Figure 2 illustrates an example of a message reconstruction process. In this example, the receiver does not receive a version m_2' of the original share m_2. The example also assumes that the secret sharing scheme in use does not require m_2' to reconstruct m.

NOTE In practice, a malicious share-holder may also try to affect the reconstruction by sending a value m_i' that does not equal m_i. Some secret sharing schemes are capable of detecting if one or more of the share-holders sent a fraudulent share.

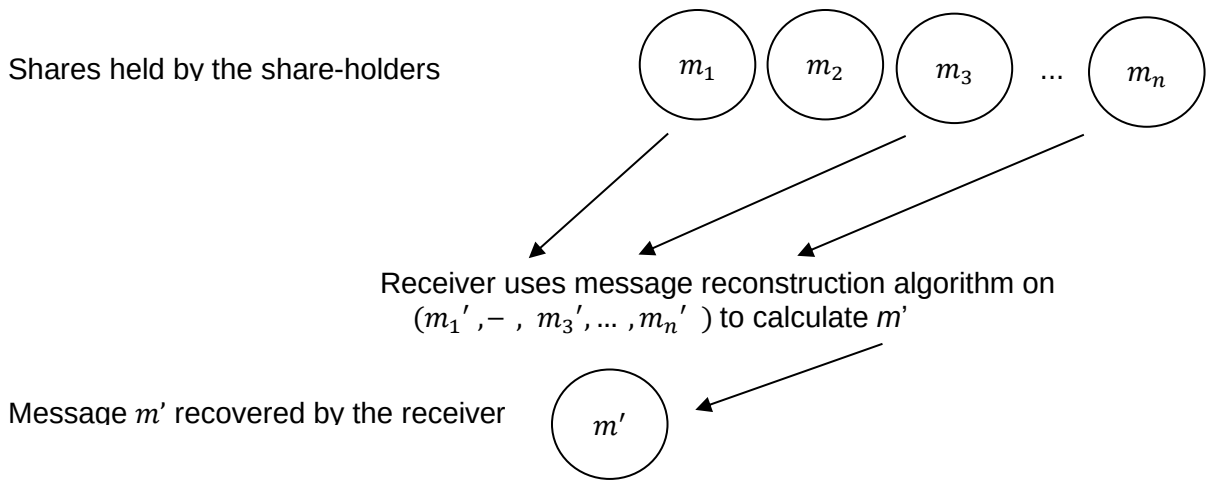


Figure 2 — Example of a message reconstruction process

5 Properties of secret sharing schemes

5.1 Fundamental requirements

5.1.1 Overview

The two fundamental requirements that shall be met by secret sharing schemes in ISO/IEC 19592 (all parts) are message confidentiality and message recoverability. The message confidentiality requirement ensures the secrecy of the shared message and the message recoverability requirement ensures availability.

5.1.2 Message confidentiality

Message confidentiality for a secret sharing scheme means that if a set of share-holders have a subset of shares that is insufficient for reconstruction, then it is infeasible for these share-holders to learn anything about the message (apart from, perhaps, its length).

NOTE For example, consider a user who uses a secret sharing scheme to divide a confidential file into a number of pieces and uploads each piece to a separate server. The secret sharing scheme instantiated by the user ensures that all pieces of the file are needed to reconstruct it. This means that only a full set of servers is authorized to recover the messages, and no incomplete subset of servers can learn anything about the confidential file from the shares they possess.

5.1.3 Message recoverability

Message recoverability for a secret sharing scheme means that if a set of share-holders have a subset of shares that is sufficient for reconstruction, these share-holders can reconstruct the message by combining their shares and applying the message reconstruction algorithm.

NOTE For example, consider a company that is protecting a private key used for creating signatures using a secret sharing scheme. The secret sharing scheme is set up in a way that allows the president of the company to reconstruct the secret key alone from the shares it is holding. Furthermore, at least two vice presidents are needed to reconstruct the key and create a valid signature. In this case, the president is authorized and any subset of two vice-presidents is authorized.

5.2 Optional requirements

5.2.1 Overview

Apart from the above fundamental requirements, secret sharing schemes may exhibit other properties relating to its parameters and processes. Examples of such properties are described in 5.2.2 and 5.2.3.

5.2.2 Homomorphicity

A secret sharing scheme can support one or more homomorphic operations on the set of possible shares. In principle, homomorphic operations allow share vectors to be combined into a new share vector in such a way that the new share vector represents the result of a meaningful operation on the original message or messages.

Consider an n-party secret sharing scheme with the message space M, share space S, a message sharing algorithm *Share* and a message reconstruction algorithm *Reconstruct.* Then, let $\oplus$ be an operation defined on the elements of M and let $\otimes$ be an operation defined on share vectors in the set S_n. This secret sharing scheme is $(\oplus,\otimes)$-homomorphic if, for all messages, $m_1, m_2 \in M$,

$$Reconstruct\,[Share\,(m_1)\ \otimes\ Share\,(m_2)]\ =\ m_1 \oplus m_2$$

Such homomorphic operations are useful in applications such as secure computation, in which a device is able to perform specified computational operations on data without having access to the cleartext values of the data.

5.2.3 Verifiability

A secret sharing scheme is verifiable if the shares output by the message sharing algorithm contain auxiliary information that allow share-holders to verify that the shares are correct, i.e. that they allow for successful reconstruction. Before reconstruction, either share-holders or the receiver can use this auxiliary information to check for the correctness of the shares. Another method for verifying that a sharing is consistent is to add auxiliary information to the shares that can be used to verify the correctness of the reconstructed message but not the shares used in the reconstruction process.

5.3 Other properties

5.3.1 Overview

Various operational and security characteristics of a secret sharing scheme can be measured, including those described in 5.3.2 to 5.3.4.

5.3.2 Confidentiality guarantees

The message confidentiality property for a secret sharing scheme can be based on a variety of assumptions. For example, a secret sharing scheme is information-theoretically confidential if it maintains message confidentiality regardless of the computational power of an unauthorized coalition of share-holders that is trying to access the message. Similarly, a secret sharing scheme is computationally confidential if it guarantees confidentiality against an unauthorized coalition with limited computational capabilities.

Furthermore, several levels of information-theoretic confidentiality can be distinguished. Confidentiality is perfect when absolutely no information is revealed about the message. However, a secret sharing scheme might only provide lesser confidentiality, meaning that an unauthorized coalition of parties may learn a non-zero number of message bits by combining their shares in a certain way.

5.3.3 Complexity

The complexity of an algorithm in a secret sharing scheme is the number of unit operations required to execute the algorithm. The complexity is specified separately for each algorithm related to the secret sharing scheme.

5.3.4 Information rate

The information rate is a property of a secret sharing scheme that represents the ratio of the message size to the maximum possible share size for a given message. If the message and share are of equal size, the information rate is 1 and the secret sharing scheme is called ideal.

There exist secret computationally secure secret sharing schemes, for example, various ramp schemes, where the ratio can be larger than one, because the message is larger than a single share.

Informasi pendukung terkait perumus standar

[1] Komite Teknis Perumusan SNI

Komite Teknis 35-04 Keamanan Informasi, Keamanan Siber, dan Perlindungan Privasi

[2] Susunan keanggotaan Komite Teknis Perumusan SNI

Ketua : Soetedjo Joewono
Sekretaris : Didik Utomo
Anggota :
1 Pedro Libratu Putu Wirya
2 Zaenal Arifin
3 Wisnoe Prasetyo Pribadi
4 Bisyron Wahyudi
5 Satriyo Wibowo
6 Sarwono Sutikno
7 Chandra Yulistia
8 Pratama Dahlian Persadha
9 Sugi Guritman
10 Bety Hayat Susanti
11 Sari Agustini Hafman

[3] Konseptor Rancangan SNI

Gugus Kerja 2 Kriptografi dan Mekanisme Keamanan – Komtek 35-04:

Ketua : Sari Agustini Hafman
Wakil Ketua : Sugi Guritman
Sekretaris : Novita Angraini
Anggota :
1. Pinuji Prasetyaningtyas
2. Bety Hayat Susanti
3. Dory Marselly
4. Afifah
5. Fadila Paradise
6. Freddy Ajax Pratama

[4] Sekretariat pengelola Komite Teknis Perumusan SNI

Direktorat Kebijakan Teknologi Keamanan Siber dan Sandi
Badan Siber dan Sandi Negara (BSSN)