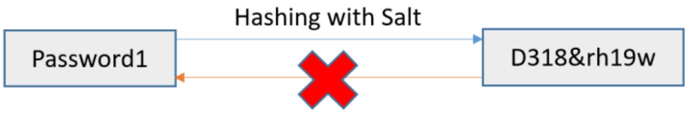
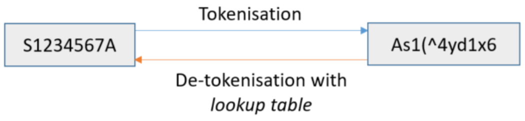
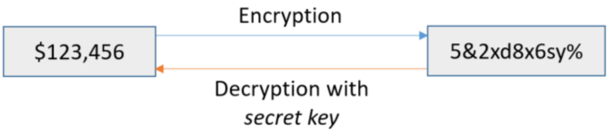
SMART NATION & DIGITAL GOVERNMENT OFFICE

250, North Bridge Road
#05-01, Raffles City Tower
Singapore 179101
smartnation.sg

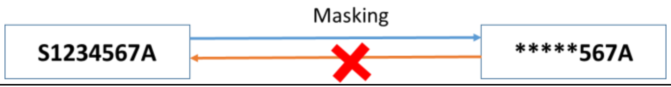
FACTSHEET – TECHNICAL MEASURES

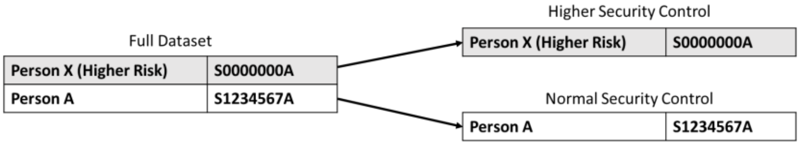
1. Hashing with salt	
Description:	Ensures that sensitive values cannot be *seen* or *reasonably recovered* in the event of a compromise. This measure is appropriate for data fields where the actual values need not be recovered, such as passwords. This technique will replace sensitive values with an algorithmically derived value that cannot be reversed easily.
Issues addressed:	Confidentiality at Storage phase
Illustrative Example:	Suppose a malicious actor manages to extract a database which stores user accounts and passwords. By applying "hashing with salt" to the passwords, the malicious actor would not be able to see the original passwords, and has no reasonable way to recover the original passwords. 

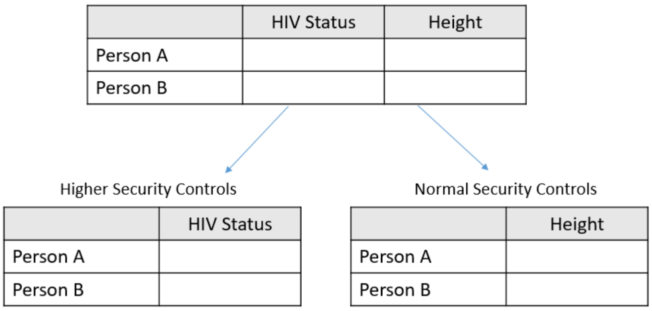
2. Tokenisation	
Description:	Ensures that identifiers cannot be *seen* in the event of a compromise. This measure replaces identifiers and attributes with a different value known only to the authorised users. This technique is appropriate for data fields where the actual values need to be recovered, such as identifiers required for service delivery.
Issues addressed:	Confidentiality at Storage phase
Illustrative Example:	Suppose a malicious actor manages to extract a database which stores NRICs. Unless the malicious actor manages to obtain the *tokens lookup table* (i.e. the full mapping of values to *tokens*), the perpetrator would not be able to see the original NRIC.  Tokenisation S1234567A As1(^4yd1x6 De-tokenisation with *lookup table*

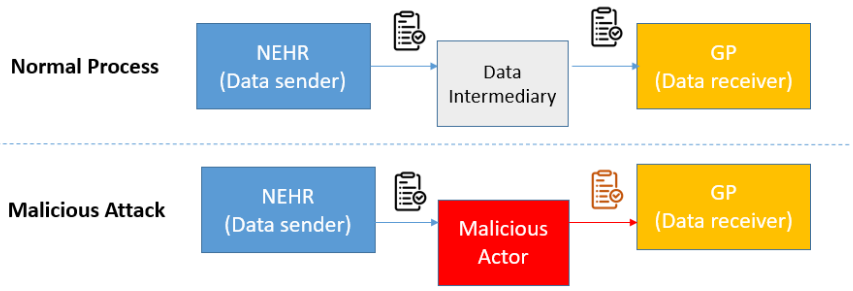
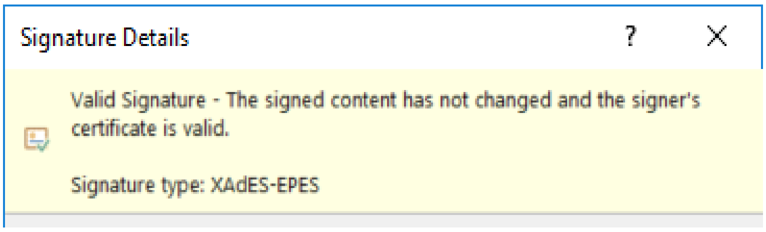
3. Field-level encryption	
Description:	Ensures that sensitive values cannot be *seen* in the event of a compromise. It involves encrypting specific data fields to hide the true value. A different secret encryption key is to be used for each field. The underlying technique of field-level encryption achieves the same function as Tokenisation, but is applied to different elements in the database (i.e. identifiers vs attributes). The technical implementation of field-level encryption uses a *mathematical encryption function* instead of a *lookup table* for tokenisation.
Issues addressed:	Confidentiality at Storage phase
Illustrative Example:	Suppose a malicious actor manages to extract a database which stores income. Unless the malicious actor manages to obtain the *secret key*, he would not be able to see the original value (income).  Encryption $123,456 5&2xd8x6sy% Decryption with *secret key*

4. Obfuscation/ masking/ removal of entity attributes	
Description:	Ensures that the exact sensitive values cannot be *seen* or *ever recovered* in the event of a compromise, although approximate or noisy values might still be seen. This involves hiding the true value of the attributes by adding noise, banding the data, or masking out portions of the value. Attributes not relevant for data usage should be removed. This measure is appropriate where the exact values are sensitive, but noisy values (that are less sensitive) are sufficient for usage and exploitation.
Issues addressed:	Confidentiality at Storage phase
Illustrative Example:	Suppose an agency wishes to send customer service agents some NRIC **for verification purposes**. They might mask the first 5 characters before sending it over, as the last 4 characters are sufficient for verification.

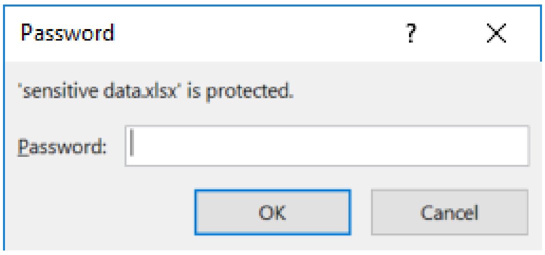
<table>
<tr><td></td><td>
</td></tr>
</table>

<table>
<tr><td colspan="2"><b>5. Dataset partitioning</b> (of entities or attributes)</td></tr>
<tr><td>Description:</td><td>Ensures that information on selected entities or attributes will not be compromised even if the larger database has been compromised. This could include protected personnel or sensitive attributes. This is done be breaking a dataset into smaller datasets by segmenting out select entities or attributes.</td></tr>
<tr><td>Issues addressed:</td><td>Confidentiality at Storage phase</td></tr>
<tr><td>Illustrative Example:</td><td><u>Example 1: Dataset partitioning of protected personnel</u>
Suppose the database of all citizens is compromised. Personal data on sensitive entities will also be compromised as part of the larger dataset compromise. By partitioning out sensitive entities, the database leak would not include these entities. Agencies can apply higher security controls to the separate database consisting of sensitive entities.


<u>Example 2: Dataset partitioning of sensitive attributes</u>
Suppose the medical database is compromised. Sensitive attributes (e.g. HIV status) will also be compromised as part of the larger dataset compromise. By partitioning out sensitive attributes, the database leak would not include these attributes.

</td></tr>
</table>

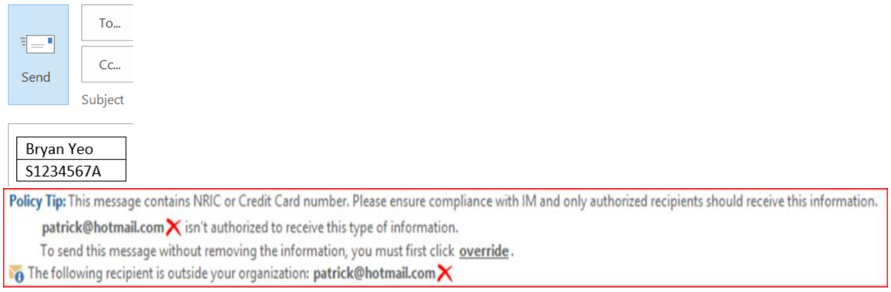
6. Data file integrity verification	
Description:	Ensures that the *receiver* gets the same file that the *original sender* intended. This is done by the original data sender providing a checksum or digital signature that confirms the integrity of a data file.
Issues addressed:	Integrity at Distribution phase
Illustrative Example:	Suppose the blood group data file of a group of patients is extracted from the National Electronic Health Record (NEHR) system and send to their doctors through a hospital. A malicious staff at the hospital might modify a person's blood group and send the modified file to doctors. This will cause the patient harm as doctors will use the wrong blood group information for treatment, even though the underlying NEHR database has not been modified.  Normal Process NEHR (Data sender) Data Intermediary GP (Data receiver) Malicious Attack NEHR (Data sender) Malicious Actor GP (Data receiver) This measure allows the doctors to verify that the data file from the NEHR has not been changed along the way, and protects against these attacks.  Signature Details Valid Signature - The signed content has not changed and the signer's certificate is valid. Signature type: XAdES-EPES File integrity verification message box

7. Password protecting and encrypting files	
Description:	Ensures that only the *receiver* with the password can access the file. This involves securing a file using encryption and password such that only authorised users can access and change the content.

Issues addressed:	Confidentiality at Distribution phase
Illustrative Example:	Suppose that an officer accidentally sends an Excel file with sensitive personal data to an unintended recipient X. Without the password, recipient X will not have access to the data in the Excel file.  Password 'sensitive data.xlsx' is protected. Password: Password prompt

8. Digital watermarking of file	
Description:	Enable investigators to trace from whom the dataset originated from in the event of a data incident. This involves adding marking information such as a cryptographic signature. The watermark information can identify the originator of the dataset, prove the authenticity of the file, and is hard to remove from the file.
Issues addressed:	Confidentiality, Integrity at Distribution phase
Illustrative Example:	Suppose an officer downloads a dataset from a central repository. The dataset will have some watermark that is unique to the officer. In the event of a data incident (e.g. the dataset being posted on online forums), investigators will be able to tell from the watermark from which officer the dataset originated. This will allow investigators to conduct better forensic analysis to identify how the incident occurred and prevent future data incidents.

9. Email data protection tool	
Description:	Ensures email senders double-check that they intended to send any email with potentially risky activities (e.g. containing sensitive data) to prevent any accidental unauthorised disclosure through email. The tools will scan for potentially risky activities (e.g. embedded files in file attachments, large numbers of users in cc lists, email contents contain identifiers such as NRICs) and require users to positively

	affirm that they intend to proceed with the potentially risky activities.
Issues addressed:	Confidentiality at Distribution phase
Illustrative Example:	Suppose an officer is sending an email with sensitive data (e.g. NRIC).  Bryan Yeo S1234567A Policy Tip: This message contains NRIC or Credit Card number. Please ensure compliance with IM and only authorized recipients should receive this information. patrick@hotmail.com ✗ isn't authorized to receive this type of information. To send this message without removing the information, you must first click override. The following recipient is outside your organization: patrick@hotmail.com ✗ When he/she clicks Send, a pop-up will appear asking for confirmation that he has intended to send such information to the recipient parties. Only upon confirmation will the email be sent out.

10. Data loss protection tools	
Description:	Prevents anomalous activities that are likely correlated with malicious activity or data incidents. Monitor computers, endpoint devices, and files for anomalous activities (e.g. unexpected downloads of large amounts of sensitive data to personal computers) and stop any unauthorised file transfers.
Issues addressed:	Confidentiality at Distribution phase
Illustrative Example:	Suppose a malicious external actor gets hold of a public officer's laptop and credentials. Using the officer's laptop, the malicious actor tries to download large amounts of sensitive data. This raises up red flags and the file transfer is automatically stopped by the data loss protection tool.

11. Volume limited and time limited data access	
Description:	Prevents officers from accessing too much data at one time, and the duration which the officer can access the data. Restrict data access when the duration and volume data access exceeds predefined limits. This can be done using the access controls features of the computer systems and in conjunction with logging and monitoring of data access.
Issues addressed:	Confidentiality at Usage phase

Illustrative Example:	Suppose a malicious actor gets hold of a public officer's laptop as well as the login and password. The malicious actor would not be able to access volumes of the data larger than the officer's predefined limit, and will not be able to access the dataset longer than the pre-set period of time (e.g. 24 hours) without reauthorisation. This mitigates the damage that the malicious actor could potentially cause.

12. Automatic Identity and Access Management (IAM) tools	
Description:	Ensures that access to the data is limited only to people authorised to do so. IAM tools automatically manage officers' identity and access rights, ensuring that only authorised persons can access data. In addition to IAM, Automatic Privileged Identity and Management (PIM) tools control, monitor, and protect user accounts which have more access and capabilities than ordinary users (e.g. administrator accounts). More stringent measures (e.g. 2-Form Factor Authentication, time-limited access) are required to protect these accounts.
Issues addressed:	Confidentiality at Usage phase
Illustrative Example:	Suppose an officer previously took on a role in Ministry of Health that authorised him to look at the infectious disease database. When that officer relinquishes that role by moving to another department, the IAM will ensure that he will no longer have any access to the infections disease dataset.

13. Enhanced logging and active monitoring of data access	
Description:	Keeps logs and analyse them to flag out anomalous activity as well as support remediation in the event of a data breach. Logging of data access to sensitive data at greater detail, such as to individual data query level. The logs should be protected from accidental or deliberate erasure, so that they can reliably show what data has been compromised, how it has been compromised and who was involved in the compromise. Active monitoring of the log files and network traffic to detect anomalies and potential malicious activities.
Issues addressed:	Confidentiality, Integrity at Usage phase
Illustrative Example:	Suppose a malicious actor performs an attack over a long period of time. The individual actions by the malicious actor at each point of

	time might not raise any red flags; however, his actions over time might be suspicious. By storing and analysing the detailed logs, these anomalous activities over time will be flagged out.