PrivBreaker VBA Macro Documentation

This document describes the required spreadsheet format, how to use the PrivBreaker VBA macros, and how the output is generated.

Overview

The PrivBreaker VBA macro suite helps identify potential privilege breaches in email data based on internal vs. external email addresses and domains.

It supports: - Parsing email fields to extract names, addresses, and domains - Identifying communication with non-internal parties - Propagating privilege flags from parent to child messages - Summarizing the results

---

Input Worksheet Format

The macro expects an **active worksheet** (e.g., `Emails`) with the following columns:

Required Columns

Column Name	Description
`BegDoc`	Unique identifier for the document/email
`ParentID`	Identifier of the parent message (blank if top-level message)
`Email From`	Sender email string (may include name and formatting)
`Email To`	Recipient email string (semi-colon delimited)
`Email CC`	CC recipient email string (semi-colon delimited)

Optional Columns (Used If Present)

Column Name	Description
`CalcPriv`	Boolean field. If present, only rows with `TRUE` will be evaluated for PrivBreak

Generated Columns

The following columns are created or overwritten by the macro: - `Email From NO`, `AO`, `DO`: Name Only, Address Only, Domain Only - `Email To NO`, `AO`, `DO` - `Email CC NO`, `AO`, `DO` - `PrivBreak`: TRUE/FALSE flag for privilege break - `PrivBreakReason`: Explanation for the flag, or blank if none

---

Internal Worksheet Format

A separate worksheet called `Internal` must be present. It defines what email addresses and domains are considered internal.

Required Columns (Starting at Row 2)

Column A (Domain)	Column B (Email)
`company.com`	`jane@company.com`
`subsidiary.org`	`legal@subsidiary.org`

Notes: - Case-insensitive: `DOMAIN.COM` will match `joe@domain.com` - Domains are used to determine if communication involves an external party - Emails override domain-level matches (i.e., if domain not in list but email is, it's considered internal)

What the Macros Do

1. `ParseEmailFields`

Parses each of the three email fields: - Extracts name, address, and domain parts from complex formats - Stores parsed data into NO, AO, DO suffix columns

2. `DetectPrivBreaks`

- For each row where `ParentID` is blank **and** (if present) `CalcPriv = TRUE`, checks:
- If any domain in `From`, `To`, or `CC` is not in `Internal` sheet
- Ensures all email addresses of that domain aren't in `Internal` sheet (overrides domain match)
- Sets `PrivBreak = TRUE` if breach detected; otherwise `FALSE`
- Populates `PrivBreakReason` with detail (tab-delimited)

3. `PropagatePrivBreaksFromParent_Fast`

- For rows with `ParentID` present
- Copies PrivBreak and sets reason as: `Inherited from parent: <ParentID>`

4. `AddPrivBreakSummary`

- Adds 2 rows at bottom:
- Count of `PrivBreak = TRUE`
- Count of `PrivBreak = FALSE`

5. `RunAllPrivBreakSteps`

Runs all the above macros in order: 1. `ParseEmailFields` 2. `DetectPrivBreaks` 3. `PropagatePrivBreaksFromParent_Fast` 4. `AddPrivBreakSummary`

Returns to original worksheet between each step to avoid focus switching.

Example Workflow

1. Prepare the `Emails` worksheet with required headers.
2. Create the `Internal` worksheet with valid internal domains and email addresses.
3. Optionally add `CalcPriv` column to control which rows are evaluated.
4. Run the `RunAllPrivBreakSteps` macro.
5. Review `PrivBreak`, `PrivBreakReason`, and summary rows.

Troubleshooting

- Ensure all required columns are spelled exactly as shown.
- Ensure `Internal` sheet exists.
- Macro will show warnings if required columns are missing.

Customization

Contact the developer to: - Add support for BCC fields - Export flagged rows to a separate sheet - Add support for logging or audit trails