國立台北大學資訊工程學系專題報告

**Access Right Controller – 權限控制器**

專題組員:葉倫全、張允隆、陳彥廷、林孟杰

專題編號: PRJ-NTPUCSIE-105-001

執行期間:105年09月 至106年05月

1. **摘要**

近年來，隨著科技的進步，到處都可以看到人類科技的結晶，但科技來自於人性，有正面，也存在著反面。其中，以電腦病毒對日常生活影響最為重大，而電腦病毒的其中一個分支勒索軟體也隨之而起，甚至成為近來無論是個人亦或是企業，最為頭痛的問題之一。因為無論是個人電腦，或是日常生活中所看到的跑馬燈，皆使用Windows系統，這也使得許多勒索軟體以Windows為目標，但市面上的防毒軟體在新型態或變種的勒索軟體出現時，仍有可能無法保護使用者資料。

對此本專題以防毒領域中的行程行為監控法為出發點，研究如何保護使用者檔案不被惡意軟體破壞，構思出檔案權限控制器(Access Right Controller，簡稱ARC)之架構，將檔案存取權限做更進一步的分類，以此保護檔案不受任意更改，並使用Windows系統所提供之minifilter驅動軟體介面實作。因此，如何使用minifilter與使用者介面實作ARC，在不影響電腦效能前提下保護使用者檔案，並同時與現存電腦程式環境共存成為本研究主要課題。

1. **簡介**

**(一) 研製背景**

由近期的網路聲量，以及google trend，又或是微軟破例為已不支援維護的作業系統發佈的系統更新檔可以得知，目前最受關注的議題之一，便是勒索軟體。勒索軟體不同於木馬病毒或是蠕蟲病毒等等，對於電腦中資料危害甚大，勒索軟體往往會向受害者要求贖金，即便電腦有備份，但只要不是在外接式硬體上進行備份，電腦便無法還原成未被加密的時間點，檔案將永遠被加密，甚至被刪除，而使用者唯一能做的決定只有重灌或是付費。

根據Net Application數據統計機構近期公布的PC作業系統市佔率報告，顯示出微軟在作業系統上的戰果依舊良好。Windows 10市佔26.28%，Wondows 8.1與Windows 8市佔8.55%，Windows XP市佔7.04%，而Windows 7依舊保持領先，市佔高達48.5%。至於其他公司的作業系統總和，例如Mac OS X與Linux，總和為12%左右。因為Windows在市佔率上有著絕對優勢，所以本次研究以Windows系統作為開發平台，實作出Access Right Controller(簡稱ARC)，研究如何有效阻擋勒索軟體。

目前市面上的防毒軟體大多使用特徵碼掃描(Signature-based detection)和主動防禦技術(Real-time protection)作為主要防禦電腦病毒的方式，但在使用特徵碼技術的基礎下，必須先擁有病毒樣本，分析得到特徵值加入資料庫後，才能有效偵測並清除該病毒，這也表示在破譯分析前，防毒軟體無法對於新的勒索軟體進行防範。

而主動防禦技術在面對勒索軟體時，通常使用加密操作作為判斷標準，試圖將勒索軟體過濾出來，但有許多軟體的功能與勒索軟體一樣都包含對檔案的加密操作，如：壓縮軟體。因此可能導致誤判並錯誤的刪除使用者的程式。

因此本專題使用行程行為監控的方法來同時抵禦已知和未知的威脅，嘗試監控系統中所有檔案操作請求並對其分類，將不安全之請求攔截並提醒使用者，以此達到保護檔案之功能。其中將請求分類的動作和分類的規則分離，提供使用者完善的設定介面和檔案請求的詳細資訊，讓使用者能動態調整分類規則，藉此符合每位使用者電腦的分類需求。

**(二) 研究目標**

 本次專題目標為設計檔案權限控制軟體並擁有三項功能，分別為(1)保護使用者資料，(2)有效對抗已知和未知的惡意軟體，以及(3)以不影響使用者既有的防毒軟體為前提，與防毒軟體共存。現今大部分防禦勒索軟體的程式以加密行為偵測為主，但使用者的檔案依舊有可能會被勒索軟體加密，這也是本專題想深入探討的原因。

**(三) 主要預期效益**

 1. 透過實作印證「作業系統」及「物件導向設計」這兩門課所學習之理論。

 2. 製作出淺顯易懂的使用者介面(User Interface)。

3. 詳細了解minifilter，並實際應用於搭載Windows作業系統之電腦平 台上。

4. 透過開發Access Right Controller(簡稱ARC)，整合系統中(Kernel mode與User mode)的溝通，達到研究目的。

5. 防護電腦資料不被勒索軟體惡意更改。

6. 透過實作與測試，驗證專題中所建立的架構在現實生活中是可以實現的。

1. **專題進行方式**
2. **系統分析與設計**

為了有效抵禦已知和未知的惡意程式，本專題以行程行為監控為基礎構思了ARC架構，主要功能包含(1)攔截系統檔案存取請求(2)提供使用者設定介面並依照使用者設定過濾檔案存取請求。

以下針對系統分析與設定進行說明，分為ARC架構概念和Windows系統下之實作方式兩大部分。

(A)ARC架構概念

ARC對每個檔案操作請求都進行監控，發現可疑或危險的請求時，會將其攔截並詢問使用者，以此達到檔案保護的功能，為了辨識可疑或危險的請求，ARC將請求進行分為以下三種。

1. 合法請求：表示安全無虞的請求，不需要對其進行阻擋，包含系統的例行請求、使用者操作所產生的請求。
2. 非法請求：表示一定需要使用者同意方能執行的請求，包含系統程式的更改請求、存取使用者重要檔案的請求。
3. 未定義請求：當操作請求不屬於合法也不屬於非法時，將其視為未定義請求。

其中合法和非法請求都是有固定格式，且可以明確描述的，而未定義請求則是只能藉由去掉合法和非法請求來取得。同時，對於每位使用者而言，所安裝的程式與檔案放置位置都不同，因此即便合法與非法請求有固定格式，對每位使用者而言仍需要不同的設定值來描述合法與非法請求，因此ARC不使用固定分類規則，而是將分類規則從分類動作分離出來，並提供使用者4種固定格式的路徑設定樣板來描述合法與非法請求，下面將敘述這4種設定模板並說明設計原因。

1. 保護路徑

保護路徑為描述非法請求之設定模板，將系統中某一路徑設定為保護路徑後，任何嘗試更改該路徑底下檔案的操作請求將視為非法請求。此設定模板之設計是用於保護使用者非常重要之檔案。

1. 共用路徑

共用路徑為描述合法請求之設定模板，將系統中某一路徑設定為共用路徑後，該路徑底下檔案操作請求將視為合法請求。此設定模板之設計是用於讓使用者設定不需要監控之例外路徑，同時也適用於描述暫存資料夾、下載資料夾內的合法請求，通常這些路徑下有很多檔案操作，但裡面的資料並不是特別重要，或是馬上就會被移除。

1. 沙盒路徑

沙盒路徑為描述合法請求之設定模板，將系統中某一路徑設定為沙盒路徑後，該路徑底下應用程式對同樣路徑底下檔案提出之更改請求將視為合法請求。此設定模板之設計是用於描述應用程式之自我更新和應用程式更改內部記錄檔和設定檔所產生的合法請求。

1. 工作區路徑

工作區路徑為描述合法請求之設定模板，當應用程式與其經常存取之檔案放置在不同之路徑下時，無法使用沙盒路徑描述此類合法請求，因此設計出此設定模板，先指定應用程式之執行檔路徑再為其指定工作區路徑，使得應用程式對工作區路徑底下之檔案所提出之更改請求視為合法請求。此設定模板最淺而易見之例子即為編譯器與程式專案檔案，編譯器之執行檔位置通常都與程式專案檔案放置位置不同，但編譯器對專案檔案提出的請求是合法請求。

當ARC發現新的檔案操作請求出現時，會取得提出該操作之程式，並取得其執行檔位置(E)，同時取得被操作之檔案路徑(F)，然後使用這兩個資訊，參考目前使用者之設定值以下列順序對請求進行分類:

1. 若F在保護路徑底下，將操作分類為非法請求。
2. 若F在共用路徑底下，將操作分類為合法請求。
3. 若E與F在同一個沙盒路徑底下，將操作分類為合法請求。
4. 若F的工作區目錄下含有E，將操作分類為合法請求。
5. 將操作分類為未定義請求。

ARC遇到合法請求時，會允許其執行；遇到非法請求時，會將其攔截並警告使用者，由使用者決定是否讓該請求執行。而遇到未定義請求時，預設的情況下ARC會讓請求通過，但會同時紀錄該程式提出未定義請求之頻率，當其頻率過高時將該程式提出之未定義請求視為非法請求，攔截並警告使用者。在此設定下，勒索軟體很容易被發現，因為勒索軟體執行加密動作時，會在短時間內提出大量的檔案更改請求。

除了頻率警告的方式外，ARC還提供另外三種處理未定義請求的方式，讓使用者根據需求套用。

1. 全部攔截：攔截所有未定義請求，讓使用者決定是否讓請求執行

當使用者希望對每一個陌生請求都擁有控制權時，可使用全部攔截的方式將未定義請求攔截再決定是否讓請求執行。

1. 全部拒絕：攔截並自動拒絕所有未定義請求

當使用者已將分類規則設定完整，希望電腦處於絕對安全的狀態時，可使用全部拒絕的方式，讓陌生的操作請求無法執行。

1. 僅記錄：讓未定義請求通過，但將請求訊息寫入紀錄檔中

當使用者對電腦所產生的合法請求格式不太清楚時，可使用僅記錄的方式處理未定義請求，將請求訊息寫入紀錄檔作為設定分類規則時的參考依據。

(B)Windows系統下之實作方式

實作ARC架構即是實現其兩個主要功能(1)攔截系統檔案存取請求(2) 提供使用者設定介面。

本專題使用Windows系統所提供之minifilter驅動軟體介面來攔截系統中的檔案存取請求。一般檔案存取請求是由User Space程式提出操作請求，經過一系列系統API呼叫、User Space/Kernel Space切換後，交由一層層的驅動程式處理，最後抵達硬體，其中會經過一個名為Filter Manager的驅動程式，而本專題所使用之minifilter即為專門設計用於掛載於Filter Manager之驅動程式，掛載之後每當檔案存取請求經過Filter Manager時，minifilter也會收到請求訊息，並且有權決定是否允許該次請求。

同時minifilter支援與User Space程式進行IPC，使得使用者可以參與每一次的請求處理，決定是否允許請求。因此除了實作Kernel Space中minifilter驅動之外，還需實作處理使用者操作和維護設定值的User Space程式，而為了使程式擁有更好之相容性，同時也為了降低程式相依性，User Space程式僅使用C++與Windows API來進行開發。

**2.** **主要困難與解決之道**

(A) 驅動程式開發不易

 實作ARC所使用之minifilter屬於驅動程式的一種，因此開發方式，部署方式皆與一般User Space之程式不同，錯誤發生時不是停止回應而是直接出現藍色死亡畫面(BSOD)，僅能依靠WinDbg和系統錯誤代碼找出bug。

(B) 定義監控範圍

 原本ARC設計中僅將寫入操作視為需要監控的對象，但是惡意程式有可能先將受保護的檔案移動到別處，進行完惡意操作後再將其恢復到原本位置，或是直接將受保護的檔案刪除，因此除了監控寫入操作外，ARC同時也監控移動、刪除操作。

(C) 監控刪除操作

 刪除檔案在Windows系統下是一個複雜的操作行為，造成刪除檔案的方法有下列方式：

1. 呼叫Windows API CreateFile使用CREATE\_NEW參數打開一個既有的檔案
2. 呼叫Windows API CreateFile使用FILE\_FLAG\_DELETE\_ON\_CLOSE參數打開既有檔案
3. 呼叫Windows API SetFileInformationByHandle將一個已經打開的檔案標記為刪除

且在檔案被標記為刪除後，檔案並不會被立刻刪除，而是要等到系統中所有指向該檔案的控制代碼(Handle)皆被關閉，檔案系統中該檔案的連結才會被真正刪除，並且在等待控制代碼全數關閉的期間內，擁有該檔案控制代碼的程式皆可以取消或重新標記刪除。因此刪除檔案連結的動作真正執行的時間點為最後一個該檔案的CloseHandle操作，並且在此時間點無法得知最後一個將檔案標記刪除的為哪一個程序，因此ARC不能使用真正的刪除動作作為監控對象，而是監控檔案開啟時CreateFile的參數以及更改檔案訊息時SetFileInformationByHandle的參數，如此一來便可將刪除操作攔截，且同時取得提出請求之程序。

**3.人員配置與職責**

 職責部分分為四部份，分別為使用者介面(User Interface)、驅動程式(minifilter)、資料夾選項功能與可疑項目功能。使用者介面由陳彥廷同學負責，驅動程式由張允隆同學負責，資料夾選項功能由林孟杰同學負責，而可疑項目功能由葉倫全同學負責。

1. **主要成果與評估**

在使用者介面(User Interface)部分，主要設計原則是簡單明瞭，讓使用者獲得最佳體驗。如圖1所示，整個程式主要可分為兩個部分，為設定部分以及程式開關與安全模式部分，可以分別進行設定以及程式運作方式。



圖1.主頁面介面

 在設定方面，除了ARC架構中的基礎功能外，還包含了特殊請求判斷的進階設定，以下為設定功能之介紹:

1. 資料夾選項：使用者可以在此使用ARC架構所提供之路徑模板描述分類規則，限制各個路徑的存取權限，條列式顯示當前路徑模板所含有的設定值，並在使用者嘗試更改設定值時，使用可展開、勾選之樹狀結構介面，且打開實際有的規則路徑會自動打勾，讓使用者能夠在選擇新增新規則或刪除舊有規則時，能夠同時知道已經設定哪些規則。



圖2.資料夾選項設定介面

1. 可疑項目:在此使用者可以清楚觀察到目前有哪些檔案操作請求被攔截下來，顯示這些請求的提出時間、攔截原因、操作行為種類、被存取之檔案等訊息，可以選擇允許或拒絕程式請求。此外，使用者可以自由選擇排序方法，例如使用時間或是名稱等等。另外，可疑項目頁面的右上角還提供了搜尋功能，可以讓使用者更精準的搜尋包含關鍵字的項目。除了這些功能以外，ARC也提供了開啟檔案位置，指的是被操作之檔案的位置，以及開啟執行檔位置，指的是正在請求操作之程式位置。這二個功能可以讓使用者在不確定操作之程式為何的情況下，可以更加仔細觀察並判斷。



圖3.可疑項目介面

1. 進階設定:在此使用者可以依照個人需求，對ARC進行部分規則更改，同時提供一些自動化選項，讓未定義請求發生時能更符合使用者所想要的使用環境。



圖4.進階設定介面

 說明完功能後，將探討程式的實用性:

 對於ARC，我們進行了數百次的測試，測試用的勒索軟體不僅種類繁複，且皆是目前最新之版本，其中還包括了五月初肆虐全球的wannacry。但測試結果證明，無論是哪一種勒索軟體，至目前為止，都無法成功躲避ARC之偵測，加密資料。

除了上述以外，ARC的另一項優勢在於幾乎不太占用系統資源，ARC開啟監控時，使用者幾乎不會感受到系統延遲，且記憶體使用量也低於20MB。

1. **結語與展望**

本次的專題研究中，成功了實踐了Access Right Controller(簡稱ARC)的架構，同時也成功遏止了勒索軟體。使用ARC取代防毒軟體對於勒索軟體的不穩定性，並使得電腦更加安全。希望藉由此次專題研究與開發，提供使用者一個更佳的防禦勒索軟體的方式，讓使用者更安心的使用電腦。

1. **銘謝**

特別感謝指導教授的指導以及提供資料給我們使用的網站。

1. **參考文獻**

Net Application , “Desktop Operating System Market Share”, **NETMARKETSHARE** (2017) from: https://www.netmarketshare.com/operating-system-market-share.aspx?qprid=10&qpcustomd=0

 Mircosoft MSDN

 (https://msdn.microsoft.com/zh-tw/default.aspx)