## 5.3 Component dependancy

Info

Er zijn verschillende ideeën over hoe software architectuur geïnterpreteerd en gerepresenteerd kan worden. Geen van deze ideeën zijn verkeerd en hebben hun eigen voordelen. Het boek ‘Practical Software Architecture’ gebruikt de definitie van Bass, Clements en Kazman (2012).

De software architectuur van een programma of computersysteem is de structuur of structuren van een softwaresysteem, bestaand uit softwarecomponenten, de externe zichtbare eigenschappen van deze componenten en de relaties tussen hen. (Bass, Clements en Kazman, 2012)

Een software architectuur bestaat uit softwarecomponenten wat in principe de bouwblokken zijn van een architectuur. Elke softwarecomponent heeft externe en interne eigenschappen. De externe eigenschappen zijn zichtbaar voor de andere componenten binnen het systeem. De details worden met de andere componenten gedeeld mits zij geen relatie hebben met de component in kwestie. Een softwarecomponent werkt als een blackbox en verbergt de interne eigenschappen voor de andere componenten omdat deze niet relevant zijn voor hen. Software architectuur gaat over het ontleden of petitioneren van een softwaresysteem in een verzameling van componenten. De componenten kunnen modulaire, iteratief en onafhankelijk opgebouwd worden en kunnen samenhangend een systeem vormen.

In figuur 5.1 worden de componenten van het van de web security scanner systeem geïllustreerd. De component C1 Website is een onderdeel van het systeem en deelt zijn eigenschappen, functionaliteiten en non functionaliteiten met andere componenten in het systeem. De component C1 Website kan ontleed worden in twee andere componenten C2 Header, C3 Link en indirect met C5 Parameter. Deze drie componenten kunnen recursief gebruikt worden en delen eigenschappen en relaties met het component C1. De component C4 Scan is een directe aggregaties van C1 Website. De relatie tussen hen geeft aan dat C4 Scan deel uit maakt van C1 Website. Dit betekent niet dat C1 Website niet zonder C4 kan bestaan. Dit is wel het geval voor C2 Header en C3 Link.

Figuur 5.2 illustratie van een recursive component dependancy diagram